

MONITOR GRAFICO
CARACTERES NO COLOR

NO 1985 Cr\$ 14.000

Micro Sistemas

A PRIMEIRA REVISTA BRASILEIRA DE MICROCOMPUTADORES



JOGOS: Programação
Editor de Adventures

GRAFIX 80 F/T. COMPATÍVEL ATÉ COM SEU PAPEL DE CARTA.



A GRAFIX 80 F/T é versátil. Com ela você pode facilmente imprimir formulários contínuos e folhas soltas como papéis de carta, envelopes, etiquetas, etc. Isto porque a GRAFIX 80 F/T é a única que vem equipada com trator e com cilindro de borracha, igual aos das máquinas de escrever, permitindo a impressão da primeira à última linha.



A GRAFIX 80 F/T é rápida e

silenciosa. Imprime em 80 colunas a 160 caracteres por segundo com baixo nível de ruído.

Além de tudo isso, a GRAFIX 80 F/T é compatível com todos os microcomputadores, especialmente com os compatíveis IBM-PC, imprimindo todos os softwares disponíveis (Dbase III, Lotus 1-2-3, Word, ABC, Chartstar etc.)

Conheça a GRAFIX 80 F/T. A impressora certa para seu micro pessoal ou profissional.

SCRITTA

Al. Amazonas, 832 - Alphaville
Barueri - SP - Tel. 421-1247

GRAFIX. TECNOLOGIA SEMPRE EM EVOLUÇÃO.



Nossa edição de JOGOS traz diversos programas e idéias interessantes, mesmo para aqueles usuários que preferem aprender a fazer do que jogar. Pensando neles, inserimos nesta edição, além das listagens de jogos, artigos que contêm dicas preciosas para os curiosos e autodidatas.

No artigo "A programação de jogos", nosso colaborador Giangiacomo Ponzo Neto percorre, com exemplo, os diversos elementos que devem constar de um bom jogo, dando sugestões de como melhor explorar os recursos das máquinas. Já Paolo Fabrizio Pugno colaborou neste número com um editor de adventures, que permitirá a você criar suas próprias aventuras. Nesta classe de jogos, um artigo deste tipo é especialmente aconselhável, visto serem os adventures programas em que o criador pode gerar enredos, cenários e personagens com vida própria. Pode ser uma excelente experiência você criar um jogo para seu amigo tentar! E além das inúmeras outras atrações que este número de MS lhe traz, os usuários da linha ZX Spectrum poderão aprender — e muito — sobre a manipulação de seu vídeo em Assembler e a geração de gráficos, se tiverem o trabalho de digitar o inédito monitor gráfico para o TK90X, de autoria do programador Divino C. R. Leitão, criador de diversos jogos comerciais, como o Walkirie, Q*BERT, Cavernas de marte e Pinball, pela marca Ciberne, e que já há alguns números integra a equipe de MS, na coordenação do CPD.

NOTA AOS LEITORES DE MS

• A partir desta edição, MICRO SISTEMAS adotará uma nova classificação no material publicado para a linha Sinclair. Na categoria ZX81, estarão os programas e dicas compatíveis com os micros nacionais TK82, TK85 e CP-200. Quando falarmos em ZX Spectrum, estaremos nos referindo aos programas que rodam no TS2068 e no TK90X (por hora, nosso único compatível com o Spectrum inglês). Finalmente, por linha Sinclair entenderemos o material que serve tanto para o ZX81 quando para o ZX Spectrum.

Alda Campos

P & D Sistemas Eletrônicos S.A.

Biblioteca

Micro Sistemas

ANO V - Nº 52 - JANEIRO 1986

SUMÁRIO

- 6** A PROGRAMAÇÃO DE JOGOS — Artigo de Giangiacomo Ponzo Neto, para a linha Sinclair.
- 12** ROLETA — Programa de Antônio José Ávila da Silva, para a linha TRS-80.
- 14** MONITOR GRÁFICO — Artigo de Divino C. R. Leitão, para a linha ZX Spectrum.
- 20** CARACTERES NO COLOR — Artigo de Cláudio Costa.
- 24** EDITOR DE ADVENTURES — Programa de Paolo Fabrizio Pugno, para a linha ZX Spectrum.
- 50** ALTA RESOLUÇÃO EM ASSEMBLER (II) — Artigo de Evandro Mascarenhas de Oliveira, para a linha Apple.
- 56** CURSO DE FORTH (IV) — Série de autoria do professor Antonio Costa.

BANCO DE SOFTWARE

- 30 THUNDERBIRD
- 32 FUGA CÔSMICA
- 36 ÓRBITA
- 38 BATALHA NAVAL
- 40 OLHADINHA
- 42 POLARIS
- 44 O PEIXE GULOSO
- 46 RALLY
- 48 MICRO-GOLF

SEÇÕES

4 CARTAS

53 LIVROS

18 BITS

54 DICAS

CAPA: José Herádio Silva

Micro Sistemas

EDITORA / DIRETORA RESPONSÁVEL:
Aida Suretus Campos

ASSESSORIA TÉCNICA:
Roberto Ouito de Sant'Anna; José Eduardo Neves; Luiz Antonio Pereira; Claudio José Costa.

CPO: Ovídio C. R. Leitão, Pedro Paulo Pinto Santos.

REDAÇÃO: Graça Santos (Subeditorial); Stella Lachter-macher; Mônica Alonso Monçães; Carlos Alberto Azevedo.

COLABORADORES: Aida Naletto Jr.; Alvaro de Filippo; Amaury Moraes Jr.; André Koch Zielasko; Antonio Costa Pereira; Ari Moralo; Cláudio de Freitas Bittencourt; Evandro Mascarenhas de Oliveira; Geraldo Simonetti Bello; Heber Jorge da Silva; Ivan Camilo da Cruz; João Antônio Zuffo; João Henrique Volpini Mattos; José Carlos Niza; José Ribeiro Pena Neto; Lúcio Pareschi; Luciano Nilo de Andrade; Luis Carlos Eiras; Luiz Carlos Nandy; Marcel Gameleira de Albuquerque; Marcel Tarras de Fontoura; Maurício Costa Reis; Marcelo Renato Rodrigues; Mário José Bittencourt; Nelson Hisashi Tamura; Nelson N. S. Santos; Newton O. Braga Jr.; Paulo Sérgio Gonçalves; Rizeri Maglio; Rudolph Horner Jr.; Sérgio Veludo.

ARTE: Cláudia Quarta (coordenação); Leonardo Santos (diagramação); Maria Christina Coelho Marques (revisão); Wellington Silveira e Marco Antônio Rubim (arte-final).

ACOMPANHAMENTO GRÁFICO: Fábio da Silva

ADMINISTRAÇÃO: Janeta Sarno

PUBLICIDADE

São Paulo:
Geni dos Santos Roberto
Contatos: Paulo Gomide;
Tel: (011) 853-3229, 853-3152

Rio de Janeiro:
Elizabeth Lopes dos Santos
Contatos: Regina Gimenez; Georgina de Oliveira
Tel: (021) 262-6306

Minas Gerais:
Sidney Domingos da Silva
Rua da Bahia, 1148 - sala 131B
CEP 30.000 - Belo Horizonte

Porto Alegre:
AURORA-ASSESSORIA EMPRESARIAL LTDA.
Rua dos Andrades, 1155 - Grupo 1806 - 16º andar
Tel.: (0512) 26-0839

CIRCULAÇÃO E ASSINATURAS:
Ademar Belon Zochio (RJ)
José Antônio Alarcon (SP) - Tel.: (011) 853-3800

COMPOSIÇÃO:
Studio Alfa, Coopim.

FOTOLITO:
Ingrafoto, Reprolito e Juracy Freire

IMPRESSÃO:
JB Indústria Gráfica

DISTRIBUIÇÃO:
Fernando Chinaglia Distribuidora Ltda.
Tel: (021) 268-9112

ASSINATURAS:
No país: 1 ano: Cr\$ 140.000

NÚMEROS ATRASADOS
Cada exemplar: Cr\$ 8.000

Os artigos assinados são de responsabilidade única e exclusiva dos autores. Todos os direitos de reprodução de conteúdo da revista estão reservados e qualquer reprodução, com finalidade comercial ou não, só poderá ser feita mediante autorização prévia. Transcrições parciais de trechos para comentários ou referências podem ser feitas, desde que sejam mencionados os dados bibliográficos de MICRO SISTEMAS. A revista não aceita material publicitário que possa ser confundido com matéria redacional.



MICRO SISTEMAS é uma publicação mensal da



Análise, Teleprocessamento e Informática Editora Ltda.

Endereços:

Rua Oliveira Dias, 153 - Jardim Paulista - São Paulo/SP - CEP 01433 - Tels: (011) 853-3800 e 881-5668 (redação)

Av. Presidente Wilson, 165 - grupo 1210 - Centro - Rio de Janeiro/RJ - Tel: (021) 262-6306.

cartas

O sorteado deste mês, que receberá uma assinatura anual da revista MICRO SISTEMAS, é Ernst R. A. Bergmann, do Rio de Janeiro - RJ.

MAPA DA RDM

Tenho algumas dúvidas com relação ao interessante artigo Apple: O Mapa da ROM, encerrado em MS nº 46. São elas:
a) Conste que o interpretador está na área \$0000-\$07FF; está correto?
b) Nas linhas \$45-\$49 estão os registradores A, X, Y, P e S ou A, X, Y, S e P, respectivamente?
c) Na área usada pelo BASIC, \$0800-\$BFFF, não constou que as áreas \$0C00-\$1FFF e \$6000-\$BFFF não são usadas a que estão livres para o usuário.
Eduardo Bonaz Marques
São Paulo - SP

De acordo com nosso procedimento habitual, enviamos sua carta ao autor do artigo e recebemos a seguinte resposta:

"Verifiquei com cuidado os pontos que você ressaltou em sua carta. Com relação ao primeiro item, realmente houve falha de minha parte: o interpretador BASIC APPLE-SOFT fica entre \$D000 e \$F7FF, e não \$D7FF. A respeito dos outros dois pontos, porém, não há erro nenhum. Tudo indica que suas dúvidas com relação à ordem de armazenamento dos registradores a áreas livres na memória são originadas do Guia do Usuário do Apple. Lá, os autores se embanaram com os registradores a chegarem a conclusões erradas, invertendo o stack pointer e o registrador de flags (S e P). A ordem correta é: A, X, Y, P e S.

Sobre as áreas \$C00-\$1FFF a \$6000-\$BFFF, cuidado! Elas não estão livres para o usuário! O programa BASIC a suas variáveis começam em \$800 a muito frequentemente vão além da \$C00, chegando mesmo a atropelar as páginas gráficas de alta resolução. Do outro lado, as strings começam a ser armazenadas logo abaixo do HIMEM: e vêm descendo em direção ao início da memória. Nesta processo de fechamento da memória, o BASIC não respeita nada, nem mesmo suas páginas gráficas - você deve protegê-las através dos comandos HIMEM: e/ou LOMEM: . Já que entramos no assunto, cuidado com a explicação que é dada no livro para o LOMEM: (LOMEM: serve para definir o endereço em que vão começar a ser armazenadas as variáveis - o interpretador executa automaticamente um LOMEM: para a primeira posição livre após o programa)."
Aldo Felício Naletto Júnior
São Paulo - SP

LIVROS/BAS

No programa Livros/Bas, publicado em MS nº 36, a partir do registro 100 do Arquivo Livros, a listagem feita na tela aparecia com uma linha em branco entre cada registro listado. Para resolver este problema, fiz algumas pequenas modificações nas linhas 610 a 1560. As modificações se resumem em acrescentar, no início das duas linhas K\$=STR\$(K); e, no final de ambas, trocar K por K\$.
Sérgio Veludo
Belém - PA

SDS ADS LEITDRES

Ao estudar Assembler no CP 400, deparei-me com o problema de geração da soma e verifiquei que para produzi-lo é necessário preparar o Sel 1 e o Sel 2 e variar a porte de som (\$FF20), gerando assim sons. Meu problema reside em como controlar os seguintes parâmetros dos sons produzidos: frequência, duração e volume. Agradeço qualquer ajuda. As cartas podem ser enviadas para a Rua Arlindo Nogueira, 1165/sul, CEP 64000. Leonardo Robert de C. Braga
Teresina - PI

MS AGRADECE

Esta é a primeira vez que escrevo e MS, apesar de ser assinante há três anos. Desejo congratulá-los pelo curso de FORTH, iniciado em MS nº 49, o qual gostei muito. Desejaria agora que fosse publicado um artigo sobre os diversos compiladores FORTH. Gostaria também que fosse publicada alguma dica ou programa para destravar os programas do TK90X.
Sotero Luiz da Silva
Santos - SP

CORRESPONDÊNCIAS

Desejo entrar em contato com os Cocomaníacos de São Paulo. Correspondências para a Rua Manuel Alonso Madina nº 207, CEP 04160, Jardim Umuarama.
J. Ronaldo Bressane Júnior (São Paulo - SP)

Gostaria de trocar informações, dicas e programas com usuários do ZX Spectrum. Meu endereço é SHIN QI - 11, cj. 13, casa 07, CEP 71500.
Alexandre M. Brito (Brasília - OF)

PONTA DE VISTA

Li com muito interesse e reportagem Software: um produto nacional bruto, em MS nº 48. No box sobre produção amadora, V. Sas. dizem que as idades dos programadores variam de 16 até 50 anos. Isto não pode ser uma verdade.

Olhem para mim: sou programador amador, muito ativo e sempre estudando. Iniciei minha carreira de programador no ano de 1982, com um NE-ZB000. Aprendi BASIC como autodidata, exatamente da maneira descrita na matéria. Além disso, fabriquei, entre outras coisas, um teclado, bem como o monitor de vídeo publicado em MS nº 21. No trabalho, uso os computadores ao máximo.

Hoje tenho, além do NE-Z, um TK 2000, um RINGO, alguns filhinhos pré-programadores, uma família encantada com os programas, uma mulher com ciúmes do micro, uma memória melhor do que nunca a 64 anos bem contados.

Eu acho que, na verdade, as idades dos programadores variam de 16 (?) até 80 anos.

Ernst R. A. Bergmann
Rio de Janeiro - RJ

Envie sua correspondência para:
ATI - Análise, Teleprocessamento e Informática Editora Ltda., Av. Presidente Wilson, 165/gr. 1210, Centro, Rio de Janeiro/RJ, CEP 20030, Seção Cartas/Redação MICRO SISTEMAS.

"BOM, É QUE NÃO CAI NA MONOTONIA. AQUI MESMO NA EMPRESA EU FAÇO CURSOS DE APRIMORAMENTO"

Fernando Lopes
Técnico de Manutenção



"O PLIM PLIM DA TV PARECIA O BIP. AÍ EU SAÍA CORRENDO"

Rogério de Figueiredo
Técnico Pleno



"NUNCA DEIXAMOS UMA MÁQUINA PARADA. SEJA QUAL FOR, ONDE FOR, VAMOS CONSERTAR!"

Mauro Diniz
Auxiliar Técnico

"MINHA MULHER NÃO ACORDA MAIS COM O BIP. EU VOU E VOLTO, E ELA NEM PERCEBE"

Luís Alberto Silva
Técnico de Manutenção



CONCERTO E MANUTENÇÃO DE COMPUTADORES

021 240-4934
011 240-5899

MICROSERVICE
SUPORTE DE INFORMÁTICA

COLOQUE O SEU PROGRAMA



O MTS-IV-M RESOLVE.

MTS-IV-M é um Micro Multiusuário, admite até 6 terminais de vídeo teclado com capacidade de executar múltiplas tarefas. Digite.

MAQUIS COMPUTADORES

Rio de Janeiro - Av. Calógeras, 6-B - Sobrelajeas
Tels.: 240-4934/220-9943 - Castelo TWX (021) 30354
CEP: 20030

São Paulo - Rua Inhambu, 1.271 - Tel.: 240-5899
Moema TWX (011) 21299
CEP 04520

Dê uma chance a sua criatividade. Com um micro da linha Sinclair, algumas técnicas e bom-senso, você poderá fazer seus jogos!

A PROGRAMAÇÃO DE JOGOS

Giorgio Ponzio Neto

Neste artigo, trataremos dos elementos constituintes de um jogo e sua elaboração, a partir de conceitos básicos, procurando exemplificá-los para um melhor entendimento. Para isso, embora o artigo procure ser o mais eclético possível, usaremos o BASIC Sinclair, por ser uma linguagem de fácil conversão e uso bastante difundido.

Dividiremos o assunto em dois itens principais: Rotinas Básicas, comuns a quase todos os jogos animados e que dão a "base" do programa; e Recursos Adicionais e Sofisticações, elementos que, mesmo não sendo obrigatórios, podem enriquecer um jogo, tornando-o mais atrativo.

A primeira coisa a se pensar é, evidentemente, a idéia do jogo. Após ter algo planejado, é uma boa prática desenvolver as situações imaginadas na tela, para se ter noção de como será o visual do que imaginamos. Assim, é necessário desenhar as personagens envolvidas numa disposição semelhante a que se deverá ter, mesmo que ainda não exista movimento, o que ajudará inclusive a aprimorar o pensamento inicial. A partir daí, passaremos às Rotinas Básicas, com a descrição das mais comuns.

DESENHO DOS COMPONENTES

Existem dois tipos de desenho num jogo animado: os fixos, que formam o cenário; e os móveis, que podem sofrer deslocamentos ou sumir e/ou aparecer. Os primeiros são os mais simples para se trabalhar (o que não implica em menor complexidade), e deve-se procurar fazê-los usando o tipo de comando mais fácil da linguagem, já que devem ser definidos no início do jogo e o loop básico do programa não deve passar por eles; não importando, assim, a eventual lentidão dos comandos usados. Nestes desenhos incluem-se paisagens de fundo, solos e outras figuras que permanecerão inalteradas.

O segundo tipo, desenhos móveis, são os que mais requerem cuidados. Sua impressão, em constante mudança de posição, deve ser rápida o suficiente para não comprometer a emoção do jogo.

Temos dois modos usuais de desenhar na tela: através de comandos do tipo PRINT, PLOT, etc., ou com uso direto do arquivo de imagem, usando POKE. O primeiro é mais simples, e apresenta vantagem, em termos de velocidade, quando o que temos a imprimir, de uma só vez, não é muito pequeno (mais de quatro caracteres). Isto porque, apesar do POKE ser mais rápido, ele só imprime um caráter por vez, sendo assim necessários vários deles para formar uma figura maior.

Desenhar com PRINT ou PLOT é elementar e só vai depender de uma boa imaginação; porém, a impressão com POKE é um pouco mais complexa. A grande maioria dos micros têm o

vídeo fixamente mapeado na memória. Neste caso, o POKE leva o endereço do ponto de tela que se quer acessar. No ZX81, entretanto, a memória de vídeo tem início variável, dependendo do tamanho do programa. Para encontrar seu início, devemos proceder como no exemplo abaixo:

```
10 LET A=PEEK 16396+256*PEEK 16397+1
20 POKE A,129
30 POKE A+1,138
40 POKE A+2,131
```

O programa anterior imprime uma "nave" nas três primeiras posições da tela. A linha 10 encontra o endereço da primeira posição. Cada linha de tela tem 32 caracteres e mais um NEW-LINE (código 118) na posição 33. Para imprimir a nave na segunda linha da tela, poderíamos fazer:

```
50 POKE A+33,129
60 POKE A+34,138
70 POKE A+35,131
```

É interessante lembrar que, uma vez encontrado o valor na linha 10, esta não precisa mais ser lida, desde que não se perca a variável (no caso: A). Portanto, a linha citada só precisa figurar na inicialização do programa.

Para maiores detalhes e dicas sobre desenhos na tela, veja o artigo "Arte e Técnica na Tela do Computador" (MS nº 27).

MOVIMENTAÇÃO DE PERSONAGENS

As figuras móveis são os elementos que representam a ação do jogo. Para tal, elas devem ser impressas de forma rápida e movidas com igual preocupação.

Para ilustrar, veremos a maneira mais comum de movimento, a partir de um caso geral. Para mover uma figura em qualquer direção, da posição X para Y, teremos, basicamente, o mesmo método: dada a figura em X, primeiro a apagamos e depois a imprimimos em Y (estes dois passos podem ser trocados). Exemplificando:

```
10 LET A=0
20 LET B=0
30 PRINT AT A,B;"S"
40 LET A=A+1
50 LET B=B+1
60 PRINT AT A-1,B-1;" "
70 GOTO 30
```

Para termos movimento em outra direção e/ou sentido basta alterar os números nas linhas 40, 50 e 60. É válido notar que alguns micros não usam a formatação de tela por linha e coluna. É o caso dos TRS-80. Para o esclarecimento desta e de outras diferenças, é muito interessante ler o artigo "Dialeto BASIC" (MS nº 46).

Não se pode, em BASIC, melhorar muito a velocidade des-

se movimento. Existem, porém, algumas exceções. Por exemplo, para um movimento horizontal na linha 21 da tela, poderíamos alterar o procedimento anterior com as seguintes modificações, deletando as linhas 10 e 40 e mantendo as outras:

```
30 PRINT AT 21,B;"S"
60 PRINT AT 21,B-1;" "
```

Entretanto, para este caso podemos usar outra rotina, bem mais simples e rápida:

```
10 LET B=0
20 PRINT AT 21,B;"S"
30 LET B=B+1
40 GOTO 20
```

Agora não há necessidade de se apagar a figura anterior, pois o espaço em branco deixado na string da linha 20 já o faz. Existe ainda a possibilidade de se usar o POKE, para uma eventual aceleração da rotina.

Uma técnica de simulação de maior velocidade é a de se imprimir a figura em mais de uma posição, além de seu local de origem. Equivaleria em substituir a rotina anterior por:

```
10 LET B=0
20 PRINT AT 21,B;"S"
30 LET B=B+2
40 GOTO 20
```

Este artifício pode ser usado para obtermos diferentes velocidades para cada figura. Perceba que o número somado na linha 30 é sempre igual ao número de espaços em branco antes da figura. Neste caso só tem sentido haver espaços do lado esquerdo, já que o movimento é só para a direita. Caso a figura também se movesse para a esquerda ou em ambos os sentidos seriam necessários ainda espaços no lado direito.

Como idéia final, ilustraremos também um caso de movimento oscilatório:

```
10 PRINT AT 10,10;"■■■";AT 10,10;"SOS"
20 FOR P=1 TO 4
30 NEXT P
40 GOTO 10
```

Esse efeito dá a sensação de superposição de imagens, podendo ser utilizado para realçar textos ou figuras ou na simulação de alguns eventos, como a movimentação das hélices de um helicóptero, por exemplo. O loop entre as linhas 20 e 30 serve apenas para definir uma espera.

ORDENS DO JOGADOR

Tudo que depende do jogador no desenrolar do jogo decorre do pressionamento de teclas específicas ou através de um joystick. Daí a necessidade de comandos que interfiram no processamento sem haver parada e alteração do programa. Na verdade, o ideal é que, além dele não ter que ser alterado, ele não pare. Isso limita muito o uso do INPUT, por ele interromper o andamento do jogo. A maneira usual de interagirmos com o programa é usar o comando INKEY\$ (GET em alguns micros). Um primeiro modo de se tentar usar o INKEY\$ para duas ou mais ações, decorrendo da mesma tecla, poderia ser realizado pela seguinte rotina:

```
10 IF INKEY$="O" THEN PRINT "A";
20 IF INKEY$="O" THEN PRINT "B";
30 GOTO 10
```

Duas ações distintas (no caso, escrever A e B na tela) decorrem do pressionamento da tecla O. A rotina anterior funciona, porém havendo um inconveniente: como não há garantia de que você vai pressionar o O exatamente quando o processamento for para a linha 10, tirando o dedo da tecla na hora em que ele sair da linha 20, é impossível ter a certeza de que as ações serão cumpridas na ordem estabelecida. Isto é resolvido no exemplo abaixo:

```
10 LET AS=INKEY$
20 IF AS="O" THEN PRINT "A";
30 IF AS="O" THEN PRINT "B";
40 GOTO 10
```

Nesse caso, a hierarquia da sequência será sempre respeitada. Outro método usual de se interagir com o jogo é criar uma "bandeira" (uma variável que sinaliza alguma condição), conforme é mostrado abaixo (a bandeira é X, que fica "ligada" quando pressionamos a tecla O).

```
10 LET X=0
20 IF INKEY$="O" THEN LET X=1
30 IF X=1 THEN PRINT "A";
40 IF X=1 THEN PRINT "B";
50 GOTO 10
```

Em alguns casos, pode-se querer uma ação no jogo em decorrência de um toque em uma tecla, sem que se precise mantê-la pressionada. Isto se consegue substituindo a linha 50 por 50 GOTO 20 e acrescentando a linha 45 IF INKEY\$="I" THEN LET X=0. A bandeira agora é usada para sinalizar a condição "desligada", quando pressionamos a tecla I.

O uso do INKEY\$ é comum também nos GOTOs condicionais, como é exemplificado abaixo:

```
IF INKEY$="O" THEN GOTO XXXX
```

Neste caso, ainda, podemos usar uma bandeira:

```
IF X=0 THEN GOTO XXXX
```

A bandeira já teria sido definida por um comando INKEY\$. Um detalhe: os exemplos mostrados são simples, e poderia se perguntar se valeriam a pena os métodos mostrados após o primeiro. Porém, há que se entender que poderíamos ter várias ações complexas, decorrendo do acionamento de teclas, e não só o simples fato de se escrever A e B na tela. Nestes casos, os métodos vistos devem ser considerados criteriosamente.

TESTES DE COLISÃO

Quando um tiro atinge um avião, ou qualquer outro caso em que o cruzamento ou colisão de figuras ocorre, imediatamente o programa deve proceder a uma modificação na situação. Esta se traduz normalmente em desvios para outras partes do programa, em mudança de condições na tela, incremento ou decremento de contadores etc. Porém, o que interessa, a priori, é detectar as eventuais colisões de figuras através dos testes.

Temos dois modos básicos de fazê-los: diretamente das informações de tela ou através do armazenamento das posições das figuras em variáveis, matrizes ou mesmo em posição mais alta da memória, criando um buffer. No primeiro caso, comparam-se os caracteres das personagens envolvidas com os caracteres diversos e procede-se à ação necessária. No segundo, comparam-se as posições na tela dos elementos que colidiram com o que havia no buffer, e toma-se a decisão programada.

Usaremos um pequeno programa para exemplificar os dois casos. Um míssil surge na tela e tenta acertar uma barreira. Pelo primeiro método teremos que, a cada "passo" do míssil, testar se a posição para a qual ele progride contém algum caráter da barreira. O caráter de fim de tela também pode ser testado para dar fim ao curso do míssil. Uma alternativa a isso é desenvolver um contador dos passos do míssil, que é checado periodicamente para sabermos o fim do tiro. Vamos ao exemplo:

```
10 FOR F=0 TO 21
20 PRINT AT F,31; "■"
30 NEXT F
40 LET B=0
50 LET A=INT (RND*61)+8
60 PRINT AT 10,30;"X";AT 11,30;"X";AT 12,30;"X"
70 PRINT AT A,B;">";AT A,B+3;
80 LET B=B+1
90 IF PEEK (PEEK 16398+256*PEEK 16199)+61 THEN GOTO 12
100 IF PEEK (PEEK 16398+256*PEEK 16199)+8 THEN GOTO 120
110 GOTO 70
120 PRINT AT 10,12;"ACERTOU"
130 PRINT AT 10,30;"■";AT 11,30;"■";AT 12,30;"■"
140 GOTO 160
150 PRINT AT 10,12;"ERROU"
160 PAUSE 80
170 PRINT AT 10,12;" " ".AT A,B;" "
180 GOTO 40
```

Uma rápida análise: as linhas 40 e 50 definem parâmetros do míssil, as linhas de 10 a 30 e 60 desenharam o fim de tela e a barreira (o caráter da linha 20 é GRAPHICS + SHIFT + A e o da linha 130 é um X inverso), as linhas 70 e 80 fazem o movimento do míssil e as linhas de 110 a 180 são detalhes gerais. O que nos interessa está contido no fim da linha 70 e nas linhas 90 e 100. O fim da linha 70 contém AT A,B+3; 2

que posiciona a variável do sistema que indica o endereço do próximo PRINT na memória do vídeo. Essa variável está, no Sinclair, nos endereços 16398 e 16399. O conteúdo do endereço que ela aponta é testado nas linhas 90 e 100. Se o número encontrado for 61 (código de "X" — barreira) ou 8 (código de "█" — fim de tela), será dado o desvio previsto. Alguns micros possuem funções especiais que determinam condições de posições de tela. O uso de variáveis do sistema ou funções especiais depende muito do micro, sendo necessário um estudo mais detalhado das suas características para viabilizar o que foi mostrado.

Para ilustrar o outro método, dê BREAK, em seguida elimine as linhas 90 e 100 e o fim da linha 70, acrescentando:

```
1 DIM X(3)
2 LET X(1)=10
3 LET X(2)=11
4 LET X(3)=12
5 LET Y=30
90 IF (A=X(1) OR A=X(2) OR A=X(3)) AND B+2=Y THEN GOTO 120
100 IF B+1=Y THEN GOTO 150
```

As linhas de 1 a 5 definem o buffer que usamos para guardar as posições da barreira (o vetor X guarda as linhas e a variável Y, a coluna). A linha 90 testa a posição para a qual o míssil vai prosseguir, comparando-a com a posição da barreira. A linha 100 não precisa testar se as posições não são da barreira para detectar o erro do alvo, pois caso fossem, o processamento já teria sido desviado pela linha 90. Se a barreira fosse móvel, seu buffer deveria ser alterado a cada movimento. A grande diferença entre os métodos reside no fato de que o segundo não se importa com o que há na tela.

O uso de um ou outro método varia muito, porém na maioria das vezes, o segundo é mais lento. E um modo de se acelerar o primeiro caso é fazer só um teste na linha 90, eliminando a linha 100. Esse teste verificaria se existe algo na posição desejada, e, em caso positivo, desviaria o processamento, onde então se saberia que tipo de colisão havia ocorrido, com acerto ou não. A velocidade aumentaria, pois o teste a cada passo do míssil seria um, ao invés de dois.

CONTADORES GERAIS

Os contadores são elementos comuns a praticamente todos os tipos de jogos existentes. Eles representam o balanço geral da situação do jogador num dado momento, pois contam os seus pontos marcados, vidas restantes, número de inimigos eventualmente atingidos, tempo decorrido ou restante, recordes anteriores e muitos outros parâmetros vitais para o desenvolvimento do jogo.

A técnica em si não traz nada de complicado. Na verdade, um contador pode ser um simples incremento em uma variável do tipo LET C = C + 1, onde C representa a expressão matemática de alguma situação real na tela. Eventualmente, poderíamos ter decremento na variável (no caso da perda de uma vida, por exemplo); ou ainda, o número a somar ou subtrair poderia ser maior ou menor que 1 (por exemplo, se determinado alvo valesse 10 pontos, outro 20, etc.).

Usando o último programa visto, poderemos mostrar alguns exemplos de contadores, se incluirmos as seguintes instruções:

```
6 LET L=0
7 LET P=0
45 LET L=L+1
125 LET P=P+1
175 IF L=5 THEN GOTO 190
190 PRINT AT 10,12;"FIM",,,,TAB 2;"O MISSIL ACERTOU ";P;" VEZES"
```

Agora temos dois contadores no programa. Rode-o e veja a diferença. Observe que ele só vai fazer cinco lançamentos (devido ao contador da variável L), que é o número de mísseis. Além disso, o programa dirá quantas barreiras foram atingidas (através do contador da variável P), que é o número de acertos.

As linhas 6 e 7 definem os contadores, as linhas 45 e 125 os incrementam e a 175 faz o teste de fim de mísseis. Perceba que na linha 100, quando comparamos B + 1 com Y, estamos usando um contador (B + 1) que, quando chega ao limite (Y), ocasiona uma ordem de desvio.

Obviamente, o programa mostrado foi feito apenas para mostrar onde se localizam e como são feitas certas rotinas elementares. Para um jogo ser realmente bom, ainda deve contar com outros aspectos.

Fazer um bom programa implica não só no uso puramente computacional de instruções, mas também em técnicas gerais de aperfeiçoamento. Essas técnicas fogem ao estilo "receita de bolo", pois variam muito de acordo com o que se pretende; e é nisso que entra a subjetividade do programador ou a sua arte. Tais aspectos dão a diferença básica entre um programa e outro que, à primeira vista, fazem a mesma coisa. Agora, veremos Recursos Adicionais e Sofisticações.

VELOCIDADE

Mais do que um recurso, a rapidez é necessária para proporcionar uma ação-reação estimulante para o jogador. Existem micros mais lentos que tornam quase impossível a criação de bons jogos, por torná-los monótonos, notadamente quando forem usadas linguagens de alto nível como, por exemplo, o BASIC. Se pudermos utilizar o BASIC compilado, teremos melhores resultados. Porém, se não dispusermos deste recurso, torna-se quase obrigatório o uso da linguagem de máquina. Algo neste sentido pode ser visto no artigo "Jogue em Ritmo Assembler" (MS n.º 40). Outra técnica consiste em evitar certos procedimentos mais lentos por natureza. Uma discussão mais prolongada sobre este e outros aspectos se encontra nos artigos "Anime-se e Faça Bons Programas em BASIC" (MS n.º 23) e "Otimize seus Programas em BASIC" (MS n.º 31).

Nas rotinas apresentadas, tentamos discorrer sobre a morosidade de algumas e a vantagem de alternativas mais velozes e igualmente eficientes.

MEMÓRIA DISPONÍVEL

Embora não seja um recurso por si só, uma maior disponibilidade de memória possibilita um maior número de situações.

Por outro lado, a economia de memória é, muitas vezes, um fator limitador no desenvolvimento de jogos. O melhor exemplo disso são os adventures: quanto menor a capacidade de armazenamento, menos situações se tornam possíveis. Entretanto, como no caso da velocidade, existem determinados métodos que levam a um menor consumo de memória. Os principais seriam: evitar comentários, espaços em branco e instruções dispensáveis e usar strings em vez de números. Porém, muito mais pode ser feito. Veja o artigo "Utilização da memória de seu Sistema de Modo Mais Eficiente" (MS n.º 5), para mais dicas.

COMODIDADE

Um jogo deve ser confortável. Certas vezes, pode ser interessante sacrificar a existência de algum comando para não comprometer a facilidade de ação por parte do jogador. Exemplificando: imagine um jogo (não é fato raro...) que use os comandos básicos de um joystick para movimentos e tiro, porém que necessite, por exemplo, de um comando especial de locomoção (caso típico do hiperespaço do "Asteroids"). Ora, é terrivelmente incômodo abrir mão do joystick para pressionar uma tecla do micro. Ideal seria se todos os comandos pudessem ser acionados no primeiro. Em linguagem de máquina pode-se, com certa facilidade, comparar o pressionamento de mais de uma tecla ao mesmo tempo. Uma opção, então, seria usar o botão de tiro e mais um comando do bastão para ordens especiais, como a citada.

Se os comandos forem do teclado, há que se tomar cuida-



Quem tem o UP-1237 da Parks não tem um modem. Tem quatro.

O Modem UP-1237 é o quatro em um da Parks porque opera em transmissões assíncronas a velocidades de 1200/75 bps ou 300 bps, no padrão CCITT ou no padrão Bell.

Você vai encontrar nesse modem outros diferenciais que a Parks desenvolveu para oferecer o máximo em tecnologia:

- dispositivo de resposta automática incorporado, possibilitando a formação de redes de computadores usando comunicação via linhas telefônicas discadas;
- comutação manual modem/telefone a partir de

MATRIZ: Porto Alegre - RS - Av. Paraná, 2335 - fone (0512) 42-5500 - Tlx. (051) 1043 - **FILIAIS:** São Paulo - fone (011) 572-7171 • Curitiba - fone (041) 232-1814 • Rio de Janeiro - fone (021) 240-7443 • Recife - fone (081) 325-2123

REPRESENTANTES: Brasília - fone (061) 272-1825 • Salvador - fone (071) 241-3091 • Belo Horizonte - fone (031) 201-5800 • Vitória - fone (027) 227-3122 • Blumenau - fone (0473) 22-1700 • Uberlândia - fone (034) 234-4600 • Cuiabá - fone (065) 321-5020 • Campo Grande - fone (067) 383-5331

- chave no painel frontal, permitindo acesso aos grandes bancos de dados nacionais e internacionais;
- seleção de velocidade no painel frontal;
- facilidade de teste de enlace analógico local para diagnóstico do equipamento;
- visualização da situação de configuração e dos circuitos de interface por meio de onze indicadores luminosos no painel frontal.

Como você sabe, para a Parks o melhor é a perfeição. Por isso, você terá a segurança de um rigoroso controle de qualidade e garantia de 2 anos. Comprove.

Garantia de
2 anos

Parks
INFORMÁTICA

do com a disposição das teclas, buscando um maior conforto. Elas não devem ser adjacentes (para não as confundirmos), nem muito distantes, e deve-se procurar distribuir as mãos no teclado de forma a não sobrecarregar a tarefa de uma delas.

COMPLEXIDADE

Para jogos de muita ação, é preferível que haja até uma certa simplicidade na idéia básica, e comandos “curtos e grossos”. Como exemplo clássico temos os “Defenders” (usualmente com apenas movimentos verticais e de tiro). Por outro lado, jogos que impliquem maior raciocínio podem ser mais complexos, e alguns chegam mesmo a necessitar disso como fator de semelhança a realidade. É o caso de simuladores-de-vôo, que dispõem de inúmeros comandos, e tendo nisto um dos seus maiores atrativos.

INCREMENTOS GERAIS

Um jogo ganha muito em emoção quando traz em si um sentido de competição; mesmo que está seja dirigida contra nós mesmos (por exemplo, ao procurarmos bater nossos próprios recordes). A simples existência de um *High Score* na tela aumenta o interesse do jogador. Neste sentido, ainda podemos contar com outros recursos; bônus a cada etapa concluída com êxito; tempo extra em caso de realização de alguma tarefa difícil; elogios para os grandes feitos ou até frases lamentando más atuações.

Quanto mais difícil for alcançar um objetivo, maior interesse temos em atingi-lo. É óbvio que existem limites humanos que não podem ser ultrapassados; logo, ao se estipular, por exemplo, o número de pontos de um jogo que se necessita para ganhar tempo extra, deve haver a preocupação que estes sejam humanamente possíveis de se alcançar.

SOM

O som é, sem dúvida, um importante fator no que diz respeito à integração (envolvimento) do jogador com o que ocorre na tela. De fato, o som se presta, em primeira instância, a dar maior veracidade à simulação de uma situação qualquer no jogo, como o tiro de um canhão.

Existe, porém, um problema operacional no uso do som de forma simples, já que, sendo o processamento das instruções sequencial, não é possível que o tiro de canhão — para usar o exemplo anterior — seja visto na tela exatamente ao mesmo tempo que é ouvido seu ruído. A solução, obviamente, passa pela necessidade de se diminuir ao máximo a diferença entre uma ação e sua audição. O problema, então, fica tão mais patente, quanto mais lento o equipamento ou a linguagem usada. Instruções próximas (se possível, subsequentes) são desejáveis para criar um efeito de simultaneidade.

Outro problema é que os comandos BASIC de geração de som são, via de regra, apenas de notas musicais e não sintetizadores de voz ou geradores de ruído. Habilmente, porém, pode-se fazer uso de notas mais baixas (menores frequências) como ruídos ou de notas intercaladas com silêncio, em ritmo acelerado, para simular helicópteros, metralhadoras, etc.

Um uso mais simples e eficiente do som é como detalhe para indicações gerais: recordes batidos, início ou fim de jogo, troca de naves e outras, justamente por não requisitarem tanto a simultaneidade já citada.

Em equipamentos sem som pode ser usado um recurso ilusório, que é o de escrever na tela o ruído do que se quer ressaltar, próximo ao agente da ação. Apesar de parecer sem graça, à primeira vista, alguns BUMs e BANGs bem localizados durante um jogo empolgante podem conseguir bons resultados.

COR

Diferentemente do som, que pode ser simulado nos micros

mudos (até por rotinas especiais), a cor não admite meio termo: ou se tem ou não; e em caso de existência, seus comandos são razoavelmente fáceis de se compreender, sendo necessário apenas usar um pouco de bom senso na escolha das cores, sem exagero. Como exemplo de abuso, poderíamos citar os primeiros programas de TV coloridos, que, ávidos de demonstrar o novo recurso, desenvolviam cenários de gosto duvidoso com todas as cores do espectro misturadas em cada quadro.

Para dispor de alguma alternativa visual, os micros sem cor podem tentar uma simulação de “cinza”, como aqueles caracteres do ZX81, alternando pontos pretos e brancos (GRAPHICS+SHIFT+A ou H), recurso esse que pode ser ampliado se dispusermos de alta resolução.

RESOLUÇÃO GRÁFICA

Como a cor, este é um recurso de tela, sendo porém mais útil que o primeiro em certos casos. É a velha história: como representar um sapo ou uma galinha com um caráter apenas? Se tivermos alta resolução (e facilidade com sua operação, o que nem sempre ocorre), poderemos então desenhar nosso bicho, ponto a ponto. Sem este recurso, teremos que nos esforçar para usar, dentro do conjunto de caracteres do micro, aqueles que mais convierem ao que se tenta representar. Caímos, assim, outra vez no uso do melhor recurso: nosso bom senso. Um asterisco representando uma aranha não é tão mal assim, já uma letra “A” é lamentável...

Som, cor e resolução gráfica, em verdade, caminham juntos. Com o surgimento dos jogos “3D”, como o Zaxxon — para citar um dos mais famosos — multicoloridos, com mil sons e ótima resolução gráfica, talvez não seja exagero ver um certo romantismo-saudosismo naqueles primeiros videogames, com o Space Invaders original. Mas daqueles bichinhos em duas dimensões e preto e branco tiramos uma grande lição: quando se tem boas idéias e se usa a cabeça, os recursos não se tornam indispensáveis.

CONCLUSÕES

Tudo o que vimos não tem a pretensão de ensinar ninguém a ser um mestre na arte de fazer jogos. São apenas algumas técnicas e refinamentos que julgamos úteis, e, às vezes, indispensáveis. Contudo, aquele toque final, que vai dar a nossa marca registrada, depende só da nossa criatividade, que representa o que sentimos e pensamos a nível individual, posto que somos distintos em idéias e gostos.

Usar as rotinas apresentadas, empilhando-as em camadas estanques, seria pouco criativo e resultaria num jogo ruim. Todas as partes de um programa deste tipo devem interagir. Imagine um avião voando, a uma velocidade X qualquer, e um canhão embaixo, esperando o momento mais adequado para o tiro. Não tem o menor sentido haver um “freio” no avião quando atiramos, por causa do maior tempo que precisamos para desenhar o tiro. Esses outros detalhes devem ser levados em conta quando fizermos um jogo. Ai então podemos dizer: “mas em BASIC não dá!”. É muitas vezes não dá mesmo; e temos que recorrer ao Assembler. Mas, não fiquemos nos culpando. Afinal, as linguagens de alto nível foram criadas para, justamente, termos melhores condições de programar. Se elas são, por vezes, muito lentas e nos obrigam a voltar às linguagens primitivas, a culpa não é nossa...

Giorgio Ponzo Neto cursa Engenharia Elétrica na UFRJ, onde vem ganhando experiência em programação FORTRAN. Trabalha também em BASIC e Assembler no seu micro da linha Sinclair e utiliza ainda calculadoras programáveis.



F = pode ser gravado em fita K7
J = uso obrigatório de joystick
48 = requer 48K de RAM
64 = requer 64K de RAM

JOGOS - APPLE II
- Cr\$ 70.000 -
Sabotage (F) - evite os pára-quadistas
Sargon (F) - jogo de xadrez
Draw Poker (F) - jogo de poker
Blackjack (F) - jogo de 21 (cartas)
Checkers (F) - jogo de damas
Fast Gammon (F) - jogo de gamão
Dthello (F) - jogo de tabuleiro
Lode Runner - pegue os tesouros
Olympic Decatlon - jogos olímpicos
Space Raiders (F) - de tipo Centopéia
The Eliminator (F) - combate espacial
Night Crawler (F) - enfrente as Centopéias
Space Eggs (F/J) - caçar ovos espaciais
Hard Hat Mack - perigos na construção
Head On (F) - jogo de corrida
Chopflifer (F/J) - resgate com helicóptero
Saa Dragon - piloto o submarino
Spy's Demise (F) - jogo de agente secreto
Star Blazer - combate aéreo
Viper (F) - alimente a cobra
Cavern Creatures - penetre na caverna
Moon Patrol (F) - patrulha lunar
Ms. Pac Man (F) - a namorada do Pac Man
Mario Bros (F) - Mário contra o goiá
Buck Rogers - pilote sua nave em "Zoom"
Pitfall II - aventuras nas cavernas perdidas
Castle Wolfenstein - engane os nazistas
Beyond Castle Wolfenstein - mate Hitler
Aztec - decifre o enigma da pirâmide azteca
Stellar 7 - combate no espaço em 3D
Rabbit (F) - jogo tipo Frogger
Super Runy - combata as pragas
Alien Ambush (F/J) - jogo tipo Arcade
Out Post (F) - defesa a estação espacial
Defender (F/J) - Defender original
Dog Fight II (F) - confronto de jatos
Star Trek - Jornada nas Estrelas
Jellyfish - pilote um submarino
Swashbuckler II - duelo de espadachins
Temple Asphal - aventura no templo
Flight Simulator (F) - simulador de vôo
One on One (J) - basquete com 2 jogadores
- Cr\$ 150.000 -
Karateka - perfeita simulação de karatê.
Enfrente vários adversários e liberte a princesa do Akuma.
Conan - localize e destrua Volta em várias fases e com diferentes obstáculos.
Ghostbusters (J) - monte uma empresa de caça aos fantasmas e comece sua aventura. Baseado no filme.
Summer Games (64) - 8 modalidades completas de atletismo, com as mais emocionantes provas olímpicas de verão.

JOGOS - APPLE II
- Cr\$ 280.000 -
JOGOS - CP-300/S00
- Cr\$ 70.000 -
Flight Simulator (F) - simulador de vôo
Olympic Decatlon (F) - jogos olímpicos

JOGOS - SPECTRUM/TK 90X
- Cr\$ 60.000 -
Fighter Pilot (48) - piloto um F-15 Eagle
Flight Simulation (48) - simulador de vôo
Hungry Horace - divertido Pac-Man
Horace & Spiders - enfrente as aranhas
Horace skilling (48) - esquis montanha abaixo
Meteors - destrua os meteoros e naves
Combate Zone - batalha de tanques em 3D
TerrorDaktl 40 (48) - ação na terra perdida
Astro Blaster - defesa o império
Trader (48) - comércio e ação no espaço
Styx (48) - ande o labirinto e mate a bruxa
Tutankhamun (48) - explore a tumba
Arcadia - sobreviva na travessia fatal
Space Intruder - jogo tipo Invaders
Tomado (48) - piloto o avião. Cenário em 3D
Planetoids - aventura espacial
Match Point (48) - fantástico tênis em 3D
Caveon (48) - uma aventura no castelo
Galaxy Attack (48) - várias fases no espaço
Chess - jogo de xadrez
Android 2 (48) - mate os robôs. Cenário em 3D
Escape - saia do labirinto sem ser morto.
Lab 3D - será que você sal dessa?
Ground Attack - jogo tipo Penetrator
Delta Wing (48) - pilote um caça F-16 em 3D
Manic Miner (48) - recolha os objetos da mina
Potty Painter (48) - preencha os quadrados
Pool - fantástico jogo de sinuca

UTILITÁRIOS/APLICATIVOS - APPLE II
- Cr\$ 320.000 -
Visicalc - planilha de cálculos
Visiplot/Visitrend - gerador de gráficos
Visidex - organizador de atividades
Visifile - gerador de banco de dados
Visichedule - análise financeira PERT/CPM
Apple LOGO - compilador
Apple FORTRAN (CP/M) - compilador
Apple Mechanic - utilitário gráfico
GraFORTH - compilador
Locksmith 5.0/F - utilitário de cópia
Nibbles Away II/C3 - utilitário de cópia
The Graphics Magicians - animador gráfico
The Graphics Solution - editor de gráficos
Print Shop - gerador de impressos gráficos
- Cr\$ 450.000 -
Turbo PASCAL (CP/M) - compilador
Take 1 - superanimador gráfico

UTILITÁRIOS/APLICATIVOS - SPECTRUM/TK 90X
- Cr\$ 320.000 -
Creator - gerador de programas BASIC
Visicalc - planilha de cálculos
Scriptit (F) - processador de textos
Clone III - copiador rápido de discos
ACCEL 3/4 - compilador BASIC
Basecom - compilador BASIC
- Cr\$ 490.000 -
NEWDOS/80 2.0 - sistema operacional
DOSPLUS 3.5 - sistema operacional
Arranger II - arquivo de diretórios
Profile III - gerador de banco de dados
SuperSCRIPT - processador de textos
ZEUS - supereditor assembler
DisnDATA - super disassembler
- Cr\$ 590.000 -
Producer - gerador de programas
Super Utility 3.2 - superutilitário

UTILITÁRIOS/APLICATIVOS - APPLE II
- Cr\$ 320.000 -
Visicalc - planilha de cálculos
Visiplot/Visitrend - gerador de gráficos
Visidex - organizador de atividades
Visifile - gerador de banco de dados
Visichedule - análise financeira PERT/CPM
Apple LOGO - compilador
Apple FORTRAN (CP/M) - compilador
Apple Mechanic - utilitário gráfico
GraFORTH - compilador
Locksmith 5.0/F - utilitário de cópia
Nibbles Away II/C3 - utilitário de cópia
The Graphics Magicians - animador gráfico
The Graphics Solution - editor de gráficos
Print Shop - gerador de impressos gráficos
- Cr\$ 450.000 -
Turbo PASCAL (CP/M) - compilador
Take 1 - superanimador gráfico

UTILITÁRIOS/APLICATIVOS - SPECTRUM/TK 90X
- Cr\$ 320.000 -
Creator - gerador de programas BASIC
Visicalc - planilha de cálculos
Scriptit (F) - processador de textos
Clone III - copiador rápido de discos
ACCEL 3/4 - compilador BASIC
Basecom - compilador BASIC
- Cr\$ 490.000 -
NEWDOS/80 2.0 - sistema operacional
DOSPLUS 3.5 - sistema operacional
Arranger II - arquivo de diretórios
Profile III - gerador de banco de dados
SuperSCRIPT - processador de textos
ZEUS - supereditor assembler
DisnDATA - super disassembler
- Cr\$ 590.000 -
Producer - gerador de programas
Super Utility 3.2 - superutilitário

UTILITÁRIOS/APLICATIVOS - SPECTRUM/TK 90X
- Cr\$ 60.000 -
Fighter Pilot (48) - piloto um F-15 Eagle
Flight Simulation (48) - simulador de vôo
Hungry Horace - divertido Pac-Man
Horace & Spiders - enfrente as aranhas
Horace skilling (48) - esquis montanha abaixo
Meteors - destrua os meteoros e naves
Combate Zone - batalha de tanques em 3D
TerrorDaktl 40 (48) - ação na terra perdida
Astro Blaster - defesa o império
Trader (48) - comércio e ação no espaço
Styx (48) - ande o labirinto e mate a bruxa
Tutankhamun (48) - explore a tumba
Arcadia - sobreviva na travessia fatal
Space Intruder - jogo tipo Invaders
Tomado (48) - piloto o avião. Cenário em 3D
Planetoids - aventura espacial
Match Point (48) - fantástico tênis em 3D
Caveon (48) - uma aventura no castelo
Galaxy Attack (48) - várias fases no espaço
Chess - jogo de xadrez
Android 2 (48) - mate os robôs. Cenário em 3D
Escape - saia do labirinto sem ser morto.
Lab 3D - será que você sal dessa?
Ground Attack - jogo tipo Penetrator
Delta Wing (48) - pilote um caça F-16 em 3D
Manic Miner (48) - recolha os objetos da mina
Potty Painter (48) - preencha os quadrados
Pool - fantástico jogo de sinuca

UTILITÁRIOS/APLICATIVOS - SPECTRUM/TK 90X
- Cr\$ 70.000 -
VU 3D - utilitário gráfico. Telas em 3D
VU Calc - planilha de cálculos
VU File - arquivo de dados para uso geral
Ultra Violet - super disassembler
MCODER II - compilador BASIC
Paint Box - editor de telas gráficas
- Cr\$ 90.000 -
SoftCALC - planilha de cálculos com GRÁFICOS
PASCAL 1.6 - compilador
SpecFDRTH - compilador
ASSEMBLER - supereditor assembler
Tasword II - processador de texto com 64 colunas

ARMED PATROL (F) - patrulha de tanques
ALLEN DEFENSE (F) - proteja-se dos invasores
METEOR MISSION (F) - caçada espacial
OUTHOUSE (F) - proteja o seu banheiro
ROBOT ATTACK (F) - destrua os robôs
SEA DRAGON (F) - pilote um submarino
MISSILE ATTACK (F) - defenda sua cidade
SCARFMAN (F) - jogo tipo Pac Man
CAVERN (F) - escape vivo da caverna
PENETRATOR (F) - aprofunde-se na cavernas
ASSAULT (F) - fuja dos ágeis inimigos
DEMON SEED (F) - acabe com os enormes pássaros
COSMIC FIGHTER (F) - jogo tipo Invasores
PANIK (F) - escape dos robôs
CRAZY PAINTER (F) - pinte a tela, se puder!
CHICKEN - ajude a galinha
GALAXY INVASION (F) - tipo Invasores
SARGON (F) - jogo de xadrez
ASALO (F) - aventura gráfica
ZORK I - superaventura
ADVENTUR - 4 aventuras diferentes
FUCFUC - programa pornográfico
GHOSTS (F) - caça os fantasmas
RALLY - pegue todas as bandeiras durante um rally

PIINTER (48) - recolha as peças do computador
PIROMANIA (48) - apague os incêndios
THE PYRAMID (48) - aventura na pirâmide
3D TANK DUEL (48) - superbatalha de tanques
3D STARSTRIKE (48) - guerra nas estrelas
SON OF BLIGGER (48) - recolha os objetos
PSYTRON (48) - defenda sua base lunar, em 3D
WHEELIE (48) - pilote uma moto no subterrâneo
FALL GUY (48) - baseado no filme DURO NA QUEDA
BOOGA-BOO (48) - ajude o sapo
ARVENTURE (48) - mate o Conde Drácula
HUNCHBACK (48) - ajude o corcunda
KARIER ATTACK - combate nas Malvinas
SPY v SPY (48) - guerra dos espies
Knight Lore (48) - aventura no castelo, em 3D
Fred (48) - desvende os pengos da pirâmide
Fantastic Voyage (48) - entre no corpo humano
Airwolf (48) - baseado no seriado ÁGUIA DE FOGO
1994 (48) - uma missão no futuro
River Raid (48) - percorra de avião um rio perigoso
Blade Alley (48) - desafio espacial em 3D
Maziacs (48) - caça um tesouro no labirinto
3D Ant Attack (48) - salve a mocinha das formigas
Spectres - um jogo diferente tipo come-come
New Poker (48) - clássico jogo de poker
Androids (48) - aventura no labirinto
Chese Coke (48) - salve os humanos
PSSST - salve sua planta das pragas
Cookie - ajude o cozinheiro
Scubadive (48) - recolha as pérolas do mar
Penetrator (48) - destrua o reator
Jetset Willy (48) - recolha os objetos
Tranz Am - pegue os troféus nos EUA
Full Throttle (48) - corrida de motos, em 3D
Chequered Flag (48) - mantenha a POLE POSITION
Worse at Saa (48) - mantenha o navio fluando
Trashman (48) - ajude a manter a cidade limpa
JetPac - construa os foguetes
Selddad Attack - combata os invasores
Aquaplane (48) - uma lancha num mar de perigos
High Moon - duelo no velho oeste
Invasion Force - destrua a nave mãe
Jump Hallager (48) - pule os obstáculos
Night Gunner (48) - pilote um bombardeio
Sir Lancelot - pegue os objetos no castelo
Ghostbusters (48) - caça aos fantasmas
Blind Alley (48) - baseado no filme TRON
Molar Maul - defenda seus dentes das cáries
Galaxians - destrua as abelhas assassinas
Monster in Hell - cuidado com o inferno

Desejo receber os seguintes programas pelos quais estou enviando um cheque nominal à ATI Editora Ltda., Av. Pres. Wilson, 165 - Grupo 1210 - Centro, CEP 20.030 Rlo de Janeiro - RJ. Despesas de correlo já Incluídas.

Computador: () CP-300/CP-300 () APPLE II () SPECTRUM/TK 90X.	ITEM (s) _____	PREÇO Cr\$ _____
Com:..... K RAM. Desejo os programas em: () FITA () DISKETTE.	_____	_____
NOME: _____	_____	_____
ENO: _____	CIOADE: _____	_____
UF: _____	CEP: _____	TEL.: _____
		TOTAL Cr\$ _____

Imprimir em quatro direções ou armazenar uma tela na memória do seu micro, compatível com o ZX Spectrum, são apenas alguns dos recursos que você conseguirá com o uso deste Monitor gráfico.

Monitor gráfico

Divino Carlos R. Leitão

Os usuários da linha ZX Spectrum têm agora um programa para auxiliá-los na criação de telas de apresentação, menus e qualquer tipo de gráfico que utilize apenas os caracteres, inclusive os redefinidos.

Este *Monitor Gráfico* é uma ferramenta que permitirá o uso de alguns recursos gráficos do seu micro, diretamente no vídeo, enquanto você cria uma tela. Alguns dos comandos disponíveis são: **INK**; **PAPER**; **FLASH**; **BRIGHT** e **INVERSE**, sendo que o comando **OVER** também foi implementado, mas apenas para que o efeito possa ser visto no vídeo, pois para armazenar os dados referentes a esse comando nas telas criadas, seria necessário aumentar o tamanho das mesmas em pelo menos 704 bytes e uma das características das telas do monitor é justamente economizar memória. Porém, como você poderá ver, este comando não fará falta.

Você ainda poderá encher o vídeo com um caráter; copiar a tela na impressora; imprimir em quatro direções; inserir e suprimir caracteres; fazer um **SCROLL** para cima e para baixo; e se o seu micro é um TK90X, poderá acessar a função **UDG 2**, diretamente no Monitor.

Uma das vantagens de uma tela criada pelo Monitor, além dos recursos gráficos prontos para serem usados, é que cada tela gasta apenas 1625 bytes e já traz incluída uma rotina de impressão em **Assembler**, tendo ainda um **UDG** próprio, que não afeta o uso do **UDG** presente na memória. As telas criadas por este processo poderão ser alocadas em qualquer posição disponível da RAM e chamadas por **RAND USR**, sendo que a única preocupação do usuário será deixar livre na memória um espaço de 1625 bytes para cada tela que desejar juntar ao seu programa.

Após criar uma tela, grave-a em fita cassete usando a opção 7 do menu. Para recuperar a tela gravada, use o comando **LOAD** "CODE XXXX", onde XXXX é o endereço de início da tela e também será o endereço de chamada para imprimi-la no vídeo.

OBS.: Para chamar uma tela diretamente de uma rotina em **Assembler**, o par BC deverá ser carregado com o valor XXXX antes de ser executado o comando **CALL XXXX**, portanto não esqueça de carregar o par BC ou haverá problemas.

Há também a opção de gravar as telas no formato padrão do micro, com a opção 5 do menu, ou ainda gravar e carregar apenas o **UDG**. Com isto, você poderá criar, em fita cassete, um banco com vários **UDGs** diferentes. As opções 1 e 2 do menu servem respectivamente para iniciar a criação de uma tela ou editar a tela que estiver na memória. Cuidado ao usar a

opção 1, pois se houver alguma tela na memória, ela será apagada.

O *Monitor Gráfico* não permite que se trabalhe com telas que não tenham sido criadas por ele, pois usa um buffer próprio de trabalho com um sistema específico de armazenamento distribuído da seguinte forma: os 49 bytes iniciais contêm a rotina de impressão que acompanha cada tela, depois vem 1408 bytes, alternando caráter e atributo, ou seja, primeiro byte = **CHR\$**; segundo byte = atributos do **CHR\$** anterior, e assim sucessivamente até totalizar as 32 colunas e 22 linhas; quanto aos 168 bytes restantes, são reservados para os caracteres do **UDG** próprio que cada tela contém.

Além das rotinas em linguagem de máquina, o programa tem uma parte em **BASIC**, para facilitar o uso das rotinas padrão de gravação e **UDG 2**.

Para digitar o programa, use o "Monitor Assembler no TK90X" (publicado em MS nº 51). Inicialmente, deverá ser criada a linha 1 REM com 4800 bytes para receber os dados em hexadecimal da listagem 1. Você irá reparar que não há 4800 dados nesta listagem, isto porque os bytes restantes serão preenchidos pelo próprio programa ao ser utilizado. Digite os dados com atenção e no final entre com a listagem abaixo, usando-a para conferir o trabalho:

```
100 REM EXECUTE COM -> RUH 100
110 LET S=0
120 FOR F=2359 TO 26633 STEP 2
130 LET S=S+PEEK F-PEEK (F+1)
140 PRINT AT 10,13;F
150 NEXT F
160 IF S=4194 THEN PRINT TAB 2;
"OK! OS DADOS ESTAO CORRETOS."
STOP
170 PRINT "FLASH! TAB 51"HA
ERRO DE DIGITACAO." TAB 51"COHF
RA OS DADOS..." STOP
```

Se o resultado for negativo, use o "Monitor Assembler no TK90X" para conferir a entrada de dados e corrija os erros. Após ter certeza de que o programa foi digitado corretamente, dê o comando direto **POKE 23756,0** para proteger a linha com as rotinas em **Assembler**, e ela então irá se transformar em linha 0 e não poderá mais ser editada. Em seguida, elimine o "Monitor Assembler no TK90X" e as linhas de conferência para então entrar com a listagem 2. Antes de testar o programa, grave tudo com o comando direto: **SAVE "MG" LINE 7**.

OBS.: Se o seu micro não é um TK90X, mude a linha 11 para 11 **GOTO 9**.

COMO USAR O MONITOR

Na figura 1, você tem uma tabela com as teclas de controle do Monitor e o seu significado, sendo que algumas devem ser pressionadas em conjunto com **CAPS SHIFT(CS)** e outras com **SYMBOL SHIFT(SS)**. O cursor indicará qual tipo de impressão está sendo usada. As convenções são:

C — imprime maiúsculas ou minúsculas junto com CS; L — imprime minúsculas ou maiúsculas junto com CS; G — imprime apenas os caracteres gráficos; S — imprime apenas os símbolos gráficos; e U — imprime apenas os caracteres redefinidos.

ROTINAS DE IMPRESSÃO NO VÍDEO

Se você já programa em linguagem de máquina ou está interessado em aprender, continue a leitura pois a partir de agora serão explicadas algumas das técnicas de impressão no vídeo em **Assembler**.

Quem já utilizava linguagem de máquina nos modelos Sinclair (ZX-81), sabe que para imprimir no D-File basta colocar o código do caráter a imprimir no endereço correspondente à posição desejada do D-File. Nos compatíveis com o ZX Spectrum, a área correspondente ao D-File dos modelos ZX-81, que a partir de agora será chamada apenas de MV (Memória de Vídeo), tem 6144 posições e mais 768 bytes que correspondem à área de atributos, — **ATTR**, para simplificar — portanto a impressão em **Assembler** fica um pouco mais complicada.

Em primeiro lugar, para se usar a MV é necessário abrir antes um canal de comunicação com o processador de vídeo. A maneira mais fácil de fazê-lo é usando uma rotina da ROM. Veja como, no exemplo 1:

```
3E02 LD A,02H ;A=2 QUE E' O CANAL DA HV
CD0116 CALL 1601H ;ABRE O CANAL 2
C9 RET ;RETORNA
```

OBS.: O canal 2(02H) corresponde apenas às 22 primeiras linhas de vídeo, de 0 a 21; para se usar as linhas 22 e 23, que chamaremos de linhas de edição, o canal aberto terá que ser 0(00H) ou 1(01H). Você deverá ter bastante cuidado quando desejar imprimir nestas duas áreas do vídeo, pois, apesar de estarem em sequência, o uso das mesmas é bem distinto. Mais à frente, você verá outras explicações sobre as diferenças das mesmas.

Após abrir o canal de vídeo, podemos imprimir direto na MV e a maneira mais prática ainda é usar as rotinas da ROM, mais especificamente o conhecido (para os já iniciados em **Assembler**) **RST 10**. Para quem não conhece, o **RST 10** é um comando de um só byte que dá acesso às rotinas de impressão dos micros Sinclair. Veja o exemplo 2:

```
3E30 LD A,30H ;A=48 QUE E' CODIGO DO CHR$ 0
D7 RST 10H ;IMPRIME O CHR$ CONTIDO EM A
C9 RET ;RETORNA
```

Como pode ser visto, não é tão difícil assim, mas não se alegre antes do tempo, pois, com o que vimos até agora, o máximo que podemos conseguir é imprimir um caráter descontroladamente na MV, sem saber qual é a sua cor, fundo ou mesmo posição onde será impresso. Para controlar estes pequenos detalhes, existem os caracteres de controle, que chamaremos de **CTRL**, ao todo são 32. Na figura 2, você tem uma tabela com a indicação das funções para as quais estes caracteres podem ser utilizados na impressão.

Os códigos de 0(00H) a 3(03H) devem ser usados seguidos de **CALL 1601H** (veja exemplo 1) para abrir os canais correspondentes, sendo que o canal 3(03H) corresponde à impressora. Já os **CTRL** de 6(06H) a 23(17H) — com exceção de 7(07H); 14(0EH) e 15(0FH) que não devem ser usados na impressão — são todos usados seguidos de **RST 10**. Veja o exemplo 3:

TECLA	CS	SS	OPERACAO
1	XX		ENCHE O VIDEO COM UM CHR\$
2	XX		MUDA MODO CURSOR C (-) L
3	XX		EXIBIR PARAMETROS DO CURSOR
4	XX		EXECUTA COMANDO COPY
5 a 8	XX		MOVE CURSOR SEM ALTERAR VIDEO
9	XX		MUDA DIRECCAO DA IMPRESSAO
0	XX		CURSOR NO MODO 0
1	XX		CURSOR NO MODO 1
2	XX		DELETE MARGEM
3	XX		COMPRIME LINHA
4	XX		EXPANDE LINHA
5	XX		ROLA O VIDEO
6	XX		SCROLL SOME
7	XX		SCROLL DESCE
8 a 9	XX		ACTIVA FUNCOES EM VERMELHO
CS+SS	XX	XX	CURSOR NO MODO U (UDG)
ENTER	XX	XX	COLOCA CURSOR NA LINHA SEGUINTE

Figura 1

```
3E16 LD A,16H ;A=22 QUE E' O CTRL DE PRINT AT
D7 RST 10H ;PREPARA PARA RECEBER PARAMETROS
3E05 LD A,05H ;A=5 QUE E' O HUMERO DA LINHA
D7 RST 10H ;POSICIONA NA LINHA
3E0A LD A,0AH ;A=10 QUE E' O HUMERO DA COLUMA
D7 RST 10H ;POSICIONA NA COLUMA
C9 RET ;RETORNA
```

Após executar esta sequência, o computador estará pronto para imprimir na linha 5, coluna 10, o caráter que for ordenado. Atenção, isto só é válido se o canal da MV corresponde às linhas 0 a 21 (exemplo 1) já tiver sido aberto antes.

Para estabelecer os parâmetros de cor, brilho e outros recursos de seu micro, use os **CTRL** de 16(10H) a 21(15H). Veja o exemplo 4:

```
3E10 LD A,10H ;A=16 QUE E' O CTRL DE INK
D7 RST 10H ;PREPARA PARA RECEBER PARAMETROS
3E07 LD A,07H ;A=7 QUE CORRESPONDE A COR BRANCA
D7 RST 10H ;MUDA INK PARA BRANCO
C9 RET ;RETORNA
```

Para mudar o **PAPER**, executa-se a mesma sequência, sendo que os códigos de cor variam de 0(00H) ou preto, até 7(07H) que é a cor branca, o que corresponde à disposição das cores no teclado. Para mudar os outros parâmetros (**FLASH**; **BRIGHT**; **INVERSE** e **OVER**), usa-se antes do segundo **RST 10** apenas 1(01H) que significa ligado ou 0(00H) que significa desligado. Veja exemplo (5) com o **FLASH**:

```
3E12 LD A,12H ;A=18 QUE E' O CTRL DE FLASH
D7 RST 10H ;PREPARA PARA RECEBER PARAMETROS
3E01 LD A,01H ;A=1 QUE CORRESPONDE A LIGADO
D7 RST 10H ;MUDA FLASH PARA LIGADO
C9 RET ;RETORNA
```

Após executar o último exemplo, o caráter a ser impresso estaria com o **FLASH** ligado, ou seja, ficaria piscando, alterando seu **INK** e **PAPER**.

Os últimos exemplos valem para todos os **CTRL** que podem ser usados na impressão, basta chamar **RST 10** após o acumulador ter sido carregado com o **CTRL** desejado e depois repetir a operação tendo sido o acumulador carregado com um código específico da função, o qual pode ser o valor de uma cor, a posição de uma coluna ou o correspondente a ligar e desligar a função. A única exceção é o **CTRL** de **PRINT AT** que precisa de dois parâmetros após ser acionado, ou seja, linha e coluna, nesta ordem. Como opção para o **PRINT AT**, você pode usar a rotina da ROM a partir de 8202 (200AH), sendo que antes é necessário carregar o par BC com o valor da linha (C) e coluna (B). Veja no exemplo 6, o que acontece se usarmos esta rotina:

```
3E16 LD A,16H ;A=22 QUE E' CTRL DE PRINT AT
D7 RST 10H ;PREPARA PARA RECEBER PARAMETROS
79 LD A,C ;A=VALOR DA LINHA CONTIDO EM C
D7 RST 10H ;POSICIONA LINHA
78 LD A,B ;A=VALOR DA COLUMA CONTIDO EM B
D7 RST 10H ;POSICIONA COLUMA
C9 RET ;RETORNA
```


Magnex lança M-20



Propondo ao usuário que não jogue fora nada do que já gastou em hardware e em software compatibilizando tudo através de um novo sistema multiusuário, a Magnex lançou o M-20. O M-20 é um computador de 8 bits que aceita como terminal qualquer microcomputador, tanto de 8 quanto de 16 bits, desde que compatíveis com CP/M. O novo equipamento da Magnex também amula, ele próprio, terminais Burroughs e Honeywell-Bull. Multiusuário a multitarefa, o M-20 suporta até quatro terminais a duas impressoras, tem 320 Kb de memória e até 20 Mb em disco Winchester. Com uma produção inicial de 30 máquinas/mês, a Magnex espera se consolidar definitivamente no mercado de pequenas e médias empresas, conquistando um aumento substancial no seu parque instalado de computadores. Dependendo das necessidades de cada usuário, o preço do M-20 vai variar de 800 a 1.500 ORTN.

Promoção Scritta

A Scritta Eletrônica está promovendo junto aos proprietários de micros IBM-PC/XT (e compatíveis) a Impressoras Grafix a distribuição gratuita do livro "Conexão Grafix IBM-PC e compatíveis". Se você está interessado em receber o livro, escreva para o Departamento de Propaganda da empresa (Alameda Amazonas, 832, Alphaville — Barueri — São Paulo, CEP: 06400) e mencione o modelo do seu microcomputador, o nº da série da impressora e o nome da publicação que veiculou esta informação.

Inaugurado o Infoshopping

No dia 5 de dezembro o Rio ganhou o primeiro shopping de informática do país. São vinte e seis lojas destinadas à venda de equipamentos ou suprimentos, clubes de vídeo, software-houses, livrarias etc.

No Rio Infoshopping estão empresas conhecidas na área como a Alaph Sistemas e Métodos, Ciência Moderna e Computação, Computer Shopping Moore, Mikros, Racimec, R&T Informática e Unitel.

O shopping conta ainda com dois cinemas, que serão usados, na parte da manhã, para a realização de palestras e seminários. O Rio Infoshopping fica na Rua do Cateite, 311, Largo do Machado (próximo à Estação do Metrô).

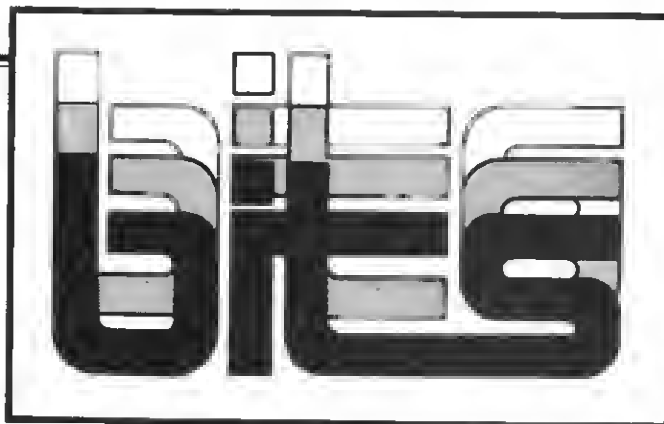
IFPS/PLUS

Através de acordo firmado com a OSS Informática, representante exclusiva da Execucom Systems Corporation no Brasil, a Intertec passou a oferecer e seus clientes o software IFPS/PLUS, entre-

Apoio ao software

Os que aspiravam muitas novidades se frustraram com o 6º Enesi — Encontro Nacional das Empresas de Serviços de Informática, promovido pela Assespro no início de dezembro em São Paulo. O evento esteve avaziado e a exposição paralela de pacotes e equipamentos também não atraiu muita gente. Na abertura do encontro, o Ministro Renato Archer, da Ciência e Tecnologia, celebrou as novas linhas especiais de crédito operantes, como o Proinfo, do BNOES, e principalmente o Programa de Apoio ao Software, da Finep, que instituiu duas modalidades de financiamento — com retorno a com participação no risco — para pequenas software-houses; system-houses e universidades.

vés do seu bureau. Trate-se de um pacote de apoio à decisão, integrado com módulos de otimização e banco de dados relacionais. Informações pelo tel.: (011) 259-2055.



Light pen para TK90X

No final deste mês já estará disponível nas lojas a light pen para se conectar ao micro TK90X, da Microdigital. Quanto ao outro periférico apresentado pela empresa na última Feira de Informática, a impressora TK Printer, ainda não há previsão de chegada ao mercado.

E por falar em mercado, a chegada dos dois novos equipamentos de linha MSX não parece ter afetado as vendas dos micros da linha Sinclair, fabricados pela Microdigital. Segundo o diretor de marketing da empresa, Jorge Luiz dos Santos, os micros da Sharp e da Gradiente deram uma força ainda maior à área da microinformática,

e pelo próprio custo destes equipamentos ele se posicionaram numa faixa diferente de que é atingida pela Microdigital. "Além do que", reforçou Jorge dos Santos, "nossos micros já contam com enorme suporte a nível técnico e de software".

O diretor de marketing da Microdigital afirmou ainda que nada impede que sua empresa venha e lançar um equipamento dessa linha no próximo Natal, principalmente se se levar em conta que as empresas que lançaram micros MSX no Brasil são novas na área, não tendo tradição em informática.

Cobra 480

A Cobra está anunciando diversos aperfeiçoamentos para o seu supermicro Cobra 480.

Estas melhorias consistem em aumentar o número máximo de linhas essenciais, acrescentar uma unidade de fita streamer de 21 Mb para back-up e adicionar uma unidade de disco Winchester de 10,51 ou 67 Mb.

Também são anunciadas novas versões dos sistemas operacionais SOO (E.00) a MUMPS (C.02), com programas utilitários para formatação e cópia de arquivos nos novos periféricos. Estes sistemas estarão disponíveis a partir de janeiro e fevereiro.

MSX Informática

A linha MSX conta agora com uma empresa especializada. Trete-se de MSX Informática Ltda. que atua tanto na comercialização desses equipamentos quanto na parte de suporte técnico e software para os mesmos. Inaugurada no final do ano, dentro da MSX Informática passa a funcionar esta também o MSX Soft Club, que conta com mais de 200 programas já disponíveis para locação. A MSX Informática funciona na rua Caiubi, nº 567, tel.: (011) 872-0730, São Paulo - Capital.

Planos para 86

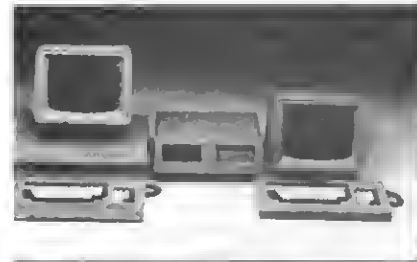
Ano novo, planos de expansão. A empresa Compumicro, que atua nos mercados do Rio de Janeiro e São Paulo, crasceu bastante ao longo de 85 e pretende diversificar suas atividades. Vários projetos têm sido estudados pela diretoria, em especial uma nova filosofia de pontos de venda e a entrada na área de treinamento.

Na área de 8 bits, — que equivale a 30% do seu volume de vendas —, a Compumicro revende basicamente UCPs de Unitrón (linha Apole); drives a impressoras da Elebra; monitores da vídeo Compo e placas da Unitrón e Logodata.

Mas é atuando no segmento de 16 bits, como representante do grupo Proceda, que a empresa tem realizado grandes negócios, comercializando principalmente para empresas estatais os micros Nexus 1600; Nexus 1684 (com Winchester de 10 Mb); PC 2001; XT 2002 e PC-PAO. Também para o mercado profissional, a Compumicro dispõe de linha de comunicação (terminais e adaptadores) da Scopus.

Para todos os produtos da Proceda, além da venda existe a opção de aluguel. Este é um contrato de 30 meses, que inclui manutenção e opção de troca, o que permite ao cliente manter a atualização tecnológica de seus sistemas e reorientar seus projetos.

Automação Industrial



Estação CAD/CAM com I-7000 PCXT

Participando de cerimônia de abertura do 2º Congresso Nacional de Automação Industrial o ministro da Ciência e Tecnologia, Renato Archer, destacou que o setor de automação deve ter o mesmo tipo de apoio que vem sendo dado à área de informática. E para isso, segundo o ministro, deve haver uma política definida que permita que os impactos da implantação da automação no mercado da mão-de-obra sejam absorvidos, a os benefícios dessa mesma implantação sejam distribuídos socialmente. Archer ressaltou

IBAM: seminário

O IBAM vai promover nos dias 23 e 24 de janeiro o seminário "Informática: tendências tecnológicas a organizacionais". Haverá turmas das 8:30 às 12:00 e das 13:00 às 17:00 hs. Maiores informações pelo tel.: (021) 266-6622.

Sorteio TK90X

Os vencedores de segunda rodada do concurso "Ganhe um micro TK90X", promovido pela Microdigital em conjunto com MICRO SISTEMAS, são Marcelo Roberto Gimenez, do Rio de Janeiro, e Carlos Mello Mesquita, de Porto Alegre.

Conexão Bolsa

A Clappy, empresa carioca especializada na venda de equipamentos, software e serviços está oferecendo um novo pacote de software: Conexão Bolsa.

Desenvolvido pela B & C Consultores, Conexão Bolsa permite a qualquer usuário dos micros Apple, TRS-80 e IBM-PC, ligar o seu equipamento aos computadores da BVRJ, possibilitando obter informações sobre o mercado de ações.

A Clappy fica na Av. Rio Branco, 12 loja. Tel: (021) 253-3395.

Cartão de referência

Foi lançado recentemente pela Ponto Editorial o cartão de referência para o micro Apple II a similares. O cartão possui 16 páginas e substitui o manual do proprietário, de 320 páginas, e aborda os principais recursos do microcomputador tais como: Integer Basic, Apolesoft, DOS 3.3 e Assembly 6502. O cartão de referência pode ser encontrado nas principais livrarias e custa Cr\$ 40 mil, e pode ser pedido também diretamente à Ponto Editorial, rua Caetés, 252 — Perdizes — CEP: 05016, tel.: (011) 864-3499 São Paulo.

Cursos BASIC

Em fevereiro, a Detamicro vai promover cursos de BASIC Básico e BASIC Avançado. O Básico começa dia 4, com turmas às segundas, quartas e sextas, em diversos horários, a o curso de BASIC Avançado terá início dia 5, com dias de aula a combinar. Informações pelo tel.: (021) 511-0395.

Arquitron

O arquiteto José Eduardo Maluf de Carvalho, colaborador de MICRO SISTEMAS a um dos primeiros a introduzir o uso do micro em sua área, está cada vez mais ligado à Informática. Prova disso é que acaba de abrir uma loja para comercialização de computadores. Mas segundo José Eduardo, na Arquitron, mais do que para o comércio de micros, a ênfase é dada a parte de serviços, tais como desenvolvimento do software para as mais variadas áreas e a criação de um clube de usuários do TK90X. A Arquitron fica na Av. Faria Lima, 1.684 sala 23, Shopping Vitrine Iguatemi, São Paulo.

Atendimento Prológica

Os usuários dos computadores pessoais da Prológica contam agora com um serviço de reparos especial. Trata-se do Atendimento Rápido Prológica, que garante o conserto quase que imediato da máquina, com prazo médio de apenas 30 minutos de espera e garantia total por 45 dias. Além disso os preços do novo serviço são fixos: CP-200, 1,8 ORTN; CP-300 e 400, 2,8 ORTN e 4 ORTN para o CP-500 modelo 021. Para meios desta ano o Atendimento Rápido Prológica deverá ser atendido à área de Sistemas e impressoras. O ARP funciona na rua Fidêncio Ramos, 302, no bairro de Vila Olímpia — São Paulo, Capital.

Novas instalações

A Maquis transfereu parte de suas dependências para a Rua Inhambu, 1271, tel.: (011) 543-3221, em Moama, São Paulo. No antigo prédio da Av. Imarés ficou apenas a fábrica.

No Rio, a Maquis também mudou de endereço. A empresa instalou seu novo show-room na Av. Calógeras, nº 6B, sobrelojas, tel.: (021) 240-4934, Centro.

Padrões brasileiros

Com o objetivo de discutir a criação de padrões brasileiros para o setor de Informática, difundir os resultados alcançados até o presente, discutir a integração desse esforço à importância da Certificação de Qualidade Industrial a colher subsídios para os trabalhos futuros, a ABNT — Associação Brasileira de Normas Técnicas —, através do seu CB-21, realizou de 18 a 20 de novembro no Hotel Glória (Rio de Janeiro), o Seminário de Normalização Técnica e Qualidade Industrial em Informática.

Constituído de vários painéis e cursos de curta duração, o Seminário destacou a importância do setor de Informática — que este ano faturará mais de US\$ 2 bilhões — para o país, enfatizou a necessidade de padronizar alguns recursos mínimos dos sistemas operacionais (chamadas de supervisão, bibliotecas, arquivos, etc.) e alguns aplicativos de uso geral, visando beneficiar a indústria, o comércio e principalmente o usuário. Os participantes também afirmaram a necessidade de que o Brasil procure soluções próprias para os seus problemas e não apenas copie programas e sistemas importados.

Outras empresas na trilha do MSX

Duas empresas do mercado de informática estudam seriamente a possibilidade de entrar na fabricação de micros da linha MSX. São elas a Dynacom e a Racimec. A Dynacom avalia com calma a hipótese de mais um produto, visto que a empresa teve problemas com as placas do micro MX-1600 — compatível com a linha Color, da Radio Shack, a com um nome bastante sugestivo, para quem pretenda investir no padrão japonês. O Color da Dynacom estava previsto para chegar às lojas no mês passado, aproveitando as vendas de fim de ano, mas devido aos tais problemas de fabricação, teve seu lançamento adiado para janeiro, segundo a diretoria de Marketing. Quanto ao MSX, poderemos ter uma surpresa, quem sabe numa dessas feiras de entre-safrá.

Já na Racimec — que não brinca em serviço e apóia-se num confortável parque instalado de seus sistemas profissionais e terminais lotéricos — as pesquisas estão bem adiantadas. Consta que já existem equipes de desenvolvimento a software dabrucadas sobre diversos modelos do MSX, somente à espera do sinal verde para ecionar os motores da fabricação. E o sinal poderá vir em breve, pois segundo o presi-

dente Simão Brayer, tudo depende da serem resolvidos os impasses de fabricação na Zona Franca de Manaus, para que haja uma "compatição limpa". Recentemente, a Gradiente teve problemas para o lançamento do micro Expert, visto que uma auditoria por parte da SEI constatou que havia plenos da fabricação do micro, contando com os incentivos da Suframa já garantidos a um projeto de videogame aprovado pelo Governo e que, afinal, nem vai ser fabricado. Oectado o "engano", a solução parece ser a mesma da Epcom, que já fabrica em Manaus o Hot Bit, porém sem contar com as regalias daquela região.

O Presidente da Racimec, contudo, parece não estar convicto desta ser a melhor solução, pois, disse ele, "pelo que conheço da burocracia e dos processos de controle, esta situação pode gerar indefinições a até facilitadas para quem já está lá". De qualquer forma, embora o impasse maior esteja na certeza de se conseguir preço competitivo, Brayer não esconde a empolgação com o produto — "Não há dúvida de que o MSX é um micro profissional" —, e como a Racimec é movida em grande parte, por seu entusiasmo, dá pra apostar.

Mesmo sendo versátil, o sistema operacional do TRS Color não possui recursos para combinar textos com gráficos em alta resolução. Porém, se você utilizar as rotinas descritas neste artigo isto será fácil.

Caracteres no Color

Cláudio Costa

Uma das maiores limitações dos micros compatíveis com o TRS Color é, sem dúvida, a impossibilidade de se combinar texto com alta resolução. Nesses equipamentos, o hardware para geração de vídeo seleciona alternativamente o modo de apresentação entre texto ou gráfico: no primeiro modo, um byte de vídeo é traduzido como um caráter; no segundo, esse mesmo byte é interpretado como um pixel ou ponto da tela gráfica (veja o artigo "Vídeo no Color", MS nº 48).

É possível concluir que tal limitação, até certo ponto, se impõe em razão do sistema operacional não ter sido dotado, a princípio, de rotinas capazes de traduzir as informações das telas gráficas na forma de caracteres. Logo, se quisermos aliar texto à alta resolução, o que temos a fazer é simplesmente escrever essas rotinas. Há várias maneiras de se fazer isso, tanto em BASIC como em linguagem de máquina.

Um método clássico é o que utiliza o comando **DRAW**. Considere uma matriz da letra A, como a ilustrada na figura 1.

Portanto, se você desejasse imprimir o caráter da figura 1, a instrução **DRAW** viria acompanhada de: **USR4D2NL4D3**, o que pode ser visto na figura 2. Este comando significa: suba cinco pontos, ande quatro para a direita, dois para baixo, quatro para a esquerda, sem atualizar a posição de impressão e, finalmente, mais três pontos para baixo (o funcionamento deste comando está claramente explicado no manual). Digite o programa abaixo e confira o resultado

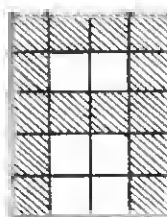


Figura 1

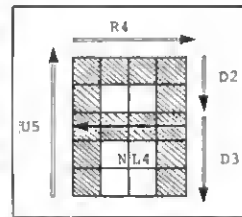


Figura 2

na tela de alta resolução:

```
10 PMODE4:COLOR5,5:PCLS:SCREEN1,
1
20 DRAW"USR4D2NL4D3"
30 GOTO30
```

Como o argumento de **DRAW** pode ser uma string, é possível estender este processo para a impressão de mensagens inteiras; basta definir as instruções **DRAW** para os caracteres e acessá-las através de uma matriz. Este método implica em estabelecer parâmetros para todo o alfabeto, trazendo porém, em contrapartida, uma grande flexibilidade. Com efeito, não precisamos nos restringir ao conjunto de caracteres convencionais do micro; podemos criar letras minúsculas, acentuação da língua portuguesa, caracteres especiais etc., tudo no formato que desejarmos.

Certos cuidados são necessários, se a intenção é criar um sistema que funcione em todos os PMODEs. Experimente, no exemplo anterior, trocar o PMODE 4 da linha 10 por PMODE 0. Rode novamente o programa, você notará que a perda de definição do desenho se deve ao fato do ponto, na baixa resolução, ser mais alto do que largo. Para funcionar em todos os PMODEs, um sistema

deve, portanto, prever as diferenças existentes entre os diversos modos gráficos.

A sub-rotina apresentada na listagem 1 imprime caracteres maiúsculos em qualquer PMODE, bastando para tal comandar **GOSUB 5**. A mensagem (máximo de 25 dígitos) deve estar contida na variável **TE\$**; **PX** define a coluna e **PY** a linha de início da impressão. Apenas o intervalo de 0 a Z foi utilizado (**CHR\$ 48** a **90**, mais um espaço). Logo, é só subtrair 47 do código ASCII de cada um dos caracteres a serem impressos para se obter a posição do desenho correspondente na matriz (linha 6). Antes, porém, são necessários alguns cálculos com o objetivo de compatibilizar a posição relativa do traço nos diversos PMODEs. Assim, **PY** deve ter um valor par, enquanto **PX** deve ser ímpar; as comparações lógicas da linha 5 verificam a consistência desses valores. O espaçamento entre os caracteres (**SP**) é calculado como sendo aproximadamente o dobro do valor da escala empregada, o qual pode ser referenciado por meio de **PEEK (233)** (linha 5). Algumas letras, como **M** e **W**, no entanto, requerem mais espaço para serem representadas; este espaço é indicado pelo parâmetro **B**, no final da instrução **DRAW** (linha 7). Note que alguns caracteres foram redefinidos: o sinal **<**, por exemplo, imprime um **ç**. Veja na figura 3 um quadro com os caracteres redefinidos na rotina. Qualquer caráter fora da faixa válida será interpretado como um espaço.

Para ver a rotina funcionando, rode o programa abaixo, pois ele imprime a

mensagem em letras ciano sobre fundo cinza (**COLOR 6, 5**) em média resolução (**PMODE 3**):

```
10 PMODE3:COLOR6,5:PCLS:SCREEN1,
1
20 TE$="CARACTERES"
30 GOSUB 5
30 GOTO30
```

A escala em que serão desenhados os caracteres pode assumir qualquer valor numa faixa de 4, 8, 12, 16 etc. Para ajustá-la, deve ser usado normalmente o parâmetro **S** do comando **DRAW**, tal como mostra o exemplo a seguir:

```
10 PMODE1:COLOR5,8:PCLS:SCREEN1,
1
20 TE$="CARACTERES"
30 GOSUB 5
30 GOTO30
```

Note, por fim, que você pode exibir numa mesma tela caracteres de diferentes formatos. Basta passar os caracteres de um PMODE para outro:

```
10 TE$="CARACTERES"
20 PMODE1:COLOR8,5:PCLS:SCREEN1,
1
30 PMODE3:COLOR6,5:PCLS:SCREEN1,
1
40 FOR T=1 TO 2000: NEXT T
50 PMODE4:SCREEN1,1:PX=81:PY=90:
GOSUB 5
60 GOTO60
```

Com base no que foi visto, você poderá adaptar a rotina de acordo com suas necessidades. Se você quiser usá-la como está, digite o comando **CLOAD** seguido da rotina e inicie a numeração de seu programa a partir de 10. Não es-

queça, nesse caso, de ajustar o parâmetro **CLEAR** da linha 1 de acordo com o programa principal.

MODIFICANDO O FORMATO DA TELA

Uma outra característica que pode passar despercebida na primeira aproximação, mas que é capaz de se tornar objeto de alguns aborrecimentos no convívio com estes micros é, certamente, a pequena capacidade de sua tela de texto, capaz de acomodar apenas 32 colunas por 16 linhas. Tal divisão, contudo, não é gratuita: como a memória do Color é dividida em páginas de 1536 bytes cada, e os 1024 bytes iniciais da primeira página estão reservados para as variáveis do sistema, sobram apenas 512 bytes para alocar a tela de texto. Como se vê, o arranjo de 32 colunas por 16 linhas é o que melhor se adapta a esta organização de memória (note que 32 x 16 perfazem exatos 512 bytes).

CARÁTER DO MICRO	CARÁTER DA ROTINA
?	.
!	C
~	A
?	.
e	.

Figura 3

Uma solução para este problema — e que resolveria ainda a incompatibilidade de texto com alta resolução — seria uma rotina que habilitasse a tela de alta resolução gráfica a funcionar também como tela de texto, mantendo-se o princípio do emprego de caracteres definidos pelo programador. Isto é possível graças a maneira como foi projetado o sistema operacional desses equipamentos, e embora tal rotina precise ser escrita em linguagem de máquina, o conceito que envolve sua operacionalidade é bastante simples.

Na verdade, a rotina do sistema operacional responsável pela impressão de um caráter, localizada em **\$A282**, executa inicialmente um **JSR** (Jump do Sub Routine, que equivale a um **GOSUB** do BASIC) para uma posição da RAM — o endereço 359 decimal, mais precisamente. Neste momento, o caráter a ser impresso está contido no registrador **A**. Esta posição contém apenas uma instrução de desvio (**JUMP**) para outro endereço da ROM, onde continua o processamento; logo, para interceptar a rotina de impressão, tudo o que temos a fazer é colocar um **JUMP** para a nossa rotina na posição apropriada da RAM, salvar os registradores naquele ponto, executar a rotina de impressão do caráter na tela gráfica, recuperar o conteúdo dos registradores e comandar um **JUMP** para

COMUNICAÇÃO ENTRE MICROS PARA TRANSFERÊNCIA DE ARQUIVOS

O **TRANSFERE** é a solução para o seu problema de ter vários micros com formatos de diskette ou sistemas operacionais incompatíveis.

Com o **TRANSFERE** você pode transferir arquivos entre dois micros através de uma ligação por cabo. O **TRANSFERE** usa as interfaces seriais dos seus micros, podendo funcionar com velocidade de transmissão de até 9600 bauds.

Na tabela ao lado você encontra as características mais importantes do **TRANSFERE**. Use-a para facilitar a sua decisão de compra, comparando o **TRANSFERE** com os concorrentes.

Computadores já suportados: **POLYMAX, EDISA, DISMAC, TELSIST, ITAUTECH, PROLOGICA, MICROTEC, SCOPUS, UNITRON, MICROENGENHO**, todos os compatíveis com o **APPLE** usando placa **CP/M** e todos os compatíveis com o **IBM PC** (consulte-nos se o seu computador não está listado, novas versões estão sendo feitas).

PREÇO: 20 ORTNs para compatíveis com o **APPLE II**,
30 ORTNs para micros **CP/M**
40 ORTNs para micros **MS-DOS**
(compatíveis com **IBM PC**)

Compare você mesmo o TRANSFERE com os concorrentes

CARACTERÍSTICAS	TRANSFERE	
Permite a transferência de arquivos entre micros CP/M e compatíveis com IBM PC	SIM	
Faz a transmissão usando um protocolo de verificação e correção automática de erros	SIM	
Transmite qualquer tipo de arquivo, como fontes de programas, textos, arquivos de dados, programas executáveis, etc.	SIM	
Permite transmitir vários arquivos com um único comando, usando a convenção de ? e * para especificar o grupo.	SIM	
A transmissão pode ser feita através de ligação direta por cabo ou através de linhas telefônicas usando modems.	SIM	
Para transferência de arquivos basta dar o comando am um dos micros ligados, o outro responde automaticamente.	SIM	
Permite acessar qualquer drive em qualquer um dos micros, podendo listar diretórios do micro local ou do remoto.	SIM	
Permite emular um terminal com protocolo TTY para ligação com CIRANDÃO, DISQUE-BOLSA, ARUANDA e outros serviços.	SIM	
Acompanha o programa um manual completo, que explica detalhadamente a sua utilização	SIM	

INTELISOFT

INTELISOFT Projeto e Desenvolvimento de Sistemas S/C Ltda
Praia do Flamengo 66 Sala 1104 CEP. 22210 Tel. (021)265 3346 - Rio de Janeiro - RJ

Listagem 1

```

1 CLEAR200: DRAW "S4": DIMCS(43): FO
RJ=1: TO43: READCS(J): NEXT J: GO TO 10
2 DATA XC$(20): NG3D5, BR3U7G2, BU6E
R3FD2GL3GD2R5, BU6ER3FDGNLFD2GL3H
, BR3U7G4R6, BU7NR5D3R4F02GL3H, BU3
R4FD6L3HU3E3R, E6UL6, XC$(4): U2EHU
, BRRE3L4HU2ER3FD2, UBU4U, UD2, XC$(
20): 806BL2D2, BU2R4BU3L4, XC$(18):
, BU9BL3, BU3R4, U, U6ER3FD3NL5D3, XC
$(33): FD2GL4
3 DATA BR4NEL3HU5ER3F, U7R4FD5GL4,
XC$(23): D4R5, U7R5BD2BDL4, XC$(20)
, BD3NL2D2, U7D3R5U3D7, BR2U7NL2NR
2D7R2, BRNHR3EU6, U7D3R2NE3F3D, NU7
R5, U7F3RE3D7B, U7F6DNU7, XC$(2U): D
5, U7R4FDGNL4, XC$(32): BL2F2, XC$(3
3): L2F4, R4EU2HL3HUER4, BR3U7L3R7B
, BU7D6FR3EU6, BU7D4F3R3EU4B
4 DATA NU7E3R3F3U7B, UESU8L5DF5D, BR
3U4NH3E3B, BU7R6G7R7
5 PX=PX OR 1: PY=PY-(PY AND 1): SP=P
EEK(233)*2+2
6 FOR J=1 TO LEN(TES): CH=ASC(MID$(T
ES, J, 1))-47: IF CH(00RCH) 43 THEN CH=
0
7 DRAW "BM" + STR$(PX) + ", " + STR$(PY)
+ CH$(CH): PX=PX+SP-2*(RIGHT$(CH$(CH
), 1) = "B"): NEXT J: RETURN
8 'VARIÁVEIS USADAS: CS, CH, J, PX,
PY, SP, TES
10 '*****

```

Listagem 2

```

10 '*****
15 ' * TEXT+ BIN
20 '*****
25 ' * Combina texto com alta
30 ' * resolucao e altera o
35 ' * formato da tela para
40 ' * 32 colunas X 24 linhas
45 '*****
50 ' * Claudio Costa * MS *
55 ' * 09/1985 *
60 '*****
65 '
70 CLEAR: NU, 256*PEEK(116)-762:
256*PEEK(116)-761: F=I+1016
75 CLS: CREFNO, I: FORB=I TO CREA
DA$(A=VAL(CHR$(A+E)): C=I+A: POKEB, A
: NEXT
80 IF C(191655) THEN PRINT@203, "E
RRO: " + POK(137, 0): IND
85 PRINT@191, "PREPARE O GRAVADOR
1: TECLA ENTER": IF INKEY$ = "TH": B
5
90 CSAVE "TEXT" + ", I, F, I
95 '
100 DATA DE, A7, EF, BD, DD, FA, 43, BD
, DD, A9, DF, A9, FE, D1, 68, EF, BD, D1, D
2, A3, BD, DD, L, I, F, D1, 68, 33, BD, D1,
5E, DF, FC, CF, 12, 64, DF, FA, C6, D4, BD
, 96, 2C, C6, 13, 97, H3, DF, B2, BD, 76, B
0, 95, CF, 33, BD, DD, 21, A6, DD, 27, 07,
43, AD, 9F, DD, DD, DD, F5, FF, D1, 68, EF

```

```

, BD, DD, 5B, 30, BD, DD, 26, BF, D1
105 '
110 DATA 68, CE, DD, CB, 34, 4D, 7E, AF
, 54, AB, BA, A7, AB, D4, F2, B2, 96, 9C, B
D, 9D, DF, AC, 96, BC, BB, 9A, 92, 9E, 8C,
DF, CE, C6, C7, C9, F2, DD, 3A, 56, 81, DC
, 26, 05, BD, 28, 17, DD, DD, 96, 77, 27, 1
1, 9E, FA, BD, A7, D3, 8L, D1, 52, D6, FF,
A7, 8D, 8C, D1, 5A, 26, F9, 96, 76, 27, DA
, CC, 0B, 04, 97, 8C, D7, BE, 8D, A9
115 '
120 DATA 56, 35, 56, 7E, 8C, F1, 34, 16
, BD, 95, 42, 9E, BA, 9F, FE, 35, 96, 7D, F
, 22, 2A, 43, 81, 9E, 27, 10, 81, 3B, 26,
D8, 34, 56, 4F, 17, DD, 9B, 35, 56, 81, 97
, 26, 02, BD, DA, B1, 40, 26, 29, 8C, AB, 7
5, 26, 24, 34, 16, BD, D3, E4, 10, 83, D2,
FF, 10, 22, 37, 5F, 1F, D1, 8C, DD, 2D, 25
, 0B, 30, 88, E0, C3, DD, E0, DD, C3
125 '
130 DATA D3, BA, DD, FE, 35, 16, B6, 2D
, 7E, AA, 1A, DD, 6F, 26, 17, 34, 56, 81, D
8, 26, 14, 4F, BD, 56, DC, FE, 3A, 2A, D3,
83, D1, E0, DD, FE, 8D, 46, 35, D6, 7E, B2
, 73, 81, DD, 26, D9, 4F, 8D, 3E, DC, FE, 4
C, 51, 20, 14, B1, DD, 25, E0, 81, 7A, 22,
E6, 8D, 2C, DC, FE, 5C, C1, DD, 25, 03, C3
, 00, E0, DD, F, 10, 93, 07, 25, 14
135 '
140 DATA DA, FE, 9E, BA, EC, 89, 01, DD
, ED, B1, 9C, FE, 26, F6, 9F, BA, BD, 95, 4
2, 96, 1C, 77, DA, 20, BB, 86, 63, 8D, 0,
C6, D7, DD, D3, FC, 1F, D1, DE, FE, 86, D7
, E6, BD, DD, B3, 26, D1, 53, E7, C4, 33, C
B, 2D, 4A, 26, F1, 39, FF, FF, FF, FF, FF,
FF, FF, F7, F7, F7, F7, F7, F7, F7, F7, E0, EB
, EB, FF, FF, FF, FF, EB, EB, C9, F1
145 '
150 DATA C9, EB, ED, F7, E1, DF, E3, FD
, C3, F7, C1, C3, FB, F7, CF, 17, D9, F7, D
7, D7, FF, 15, D8, ES, F7, F7, EF, FF, FF,
FF, 17, F7, EF, DF, DF, DF, CF, 17, F7, FB
, FD, 1D, FD, FB, F7, FF, F7, E3, 11, E3, F
7, FF, F7, F7, C1, F7, F7, FF, FF, FF,
FF, 17, F7, F7, FF, FF, F7, F7, C1, FF, FF,
FF, FF, 17, F7, FF, FF, F7, F7, C1, FF, FF,
FF, FF, 17, F7, FF, FF, F7, F7, F7, F7, F7, F7
155 '
160 DATA A, FD, FB, F7, EF, DF, DF, E7, DB
, DB, DD, DB, DB, E7, F7, F7, F7, F7, F7,
F7, E3, 13, DD, FB, 13, DF, DF, C1, E3, DD,
FD, C3, FD, DD, E3, FB, F7, F7, C1, FB, FB
, FB, E1, DF, C3, 1D, FD, DD, E3, E3, DF, D
F, C3, DD, DD, F3, C1, FD, FB, F7, EF, DF,
DF, 13, DD, DD, F3, FD, DD, F3, E3, DD, DD
, E1, FD, FD, 13, FF, F7, E7, FF, F7
165 '
170 DATA E7, FF, E7, E7, F1, E7, E7, F7
, E7, FB, F7, EF, DF, DF, F7, FB, FF, F7, C
1, FF, C1, F, FF, EF, F7, F7, FD, FB, F7,
EF, E7, DD, FB, F7, F7, F1, F7, E3, DD, FD
, ED, D5, D5, C3, F7, F7, FD, DD, C1, DD, D
0, C3, CD, ED, E3, ED, ED, C3, E3, DD, DF,
DF, DD, DD, C3, C3, CD, ED, ED, ED, C3
, C1, DF, DF, C3, DF, DF, C1, C1, DF
175 '
180 DATA DF, C3, DF, DF, DF, E1, DF, DF
, D9, DD, DD, E1, DD, DD, C1, DD, DD, D
1, C3, F7, F7, F7, F7, F7, F7, F7, F7, F7,
FD, DD, DD, E3, DD, DD, D7, CF, D7, DD, DD
, DF, DF, DF, DF, DF, D1, C1, DD, C9, D5, D
5, DD, DD, DD, DD, DD, C0, D5, D9, DD, DD, D
C1, DD, DD, DD, DD, DD, C1, C3, DD, DD, C3
, DF, DF, DF, E3, DD, DD, DD, D5, DB
185 '
190 DATA E5, C3, DD, DD, C3, D7, DD, DD
, E3, DD, EF, F7, FB, DD, E3, C1, F7, F7, F
7, F7, F7, F7, DD, DD, DD, DD, DD, E3,
DD, DD, DD, E8, E8, F7, F7, DD, DD, DD, D
5, C9, DD, DD, DD, DD, E8, F7, E8, DD, DD, D
0, DD, E8, F7, F7, F7, F7, C1, FD, FB, F7,
EF, DF, C1, C7, DF, DF, DF, DF, DF, C7, E7
, DB, DB, E7, FF, E7, FF, E3, FB, FB
195 '
200 DATA FB, FB, FB, E3, F7, E3, D5, F7
, F7, F7, F7, F7, FB, B1, FB, F7, FF, F
F, FF, FF, FF, FF, C1, FF, FF, C7, FB,

```

```

C3, BB, 17, DF, DF, D3, DD, DD, DD, C3, FF
, FF, E1, DD, DF, DD, E3, FD, FD, E1, DD, D
0, DD, E1, FF, E3, DD, C1, DF, E1, F3,
, E1, C, EF, EF, EF, FF, FD, E3, DD, E3
, 1D, C3, DF, DF, D3, DD, DD, DD, DD
205 '
210 DATA F7, FF, F7, F7, F7, F7, E3, FB
, F1, FB, 1B, FB, FD, E7, DF, DF, D8, D7, U
7, DB, DD, E7, F7, F7, F7, F7, F7, F7, F7,
FF, B9, B6, B6, B6, BE, FF, FF, D3, DD, DD
, DD, DD, FF, FF, E3, DD, DD, DD, E3, FF, D
3, ED, DD, C3, DF, DF, FF, E5, D9, DD, E1,
FD, FD, FF, FF, D3, CD, DF, DF, FF, FF,
F3, DF, E3, FD, E3, EF, CF, B3, CF
215 '
220 DATA FF, ED, F3, FF, FF, DD, DD, DD
, DD, 13, F1, FF, DD, DD, DD, EB, F7, FF, F
F, BE, B6, B6, B6, B6, FF, FF, D3, E3, F7,
EB, DD, FF, FF, DD, DD, E1, 1D, E3, FF, FF
, C1, FD, E3, DF, C1

```

o endereço originalmente apontado por aquela posição da RAM. Evidentemente, o programa não se resume apenas nisso; uma série de outras funções precisam ser implementadas, tais como: impressão do cursor, delete, backspace, execução do SCROLL automático quando a tela estiver cheia etc.

É bom saber, entretanto, que é possível escrever um programa que execute todas essas funções usando menos de 300 bytes. O maior consumo de memória, na realidade, está relacionado com a tabela de que tal programa necessita para desenhar os caracteres. Não é possível, a exemplo do que fazem os usuários do ZX81, ler a tabela de definição de caracteres do micro, porque essa tabela não está localizada na ROM do sistema operacional e sim na ROM do microprocessador de vídeo, o VDQ.

Como a definição do formato dos caracteres fica a critério do programador, não existem, a rigor, restrições quanto ao número de linhas ou colunas que podem ser utilizadas pelo novo sistema. Alguns critérios, todavia, devem ser seguidos e um deles diz respeito à legibilidade dos caracteres. Isto pode parecer óbvio, mas muitos programas comerciais desenvolvidos com o objetivo de ampliar a capacidade da tela de texto do Color Computer pecam nesse aspecto. Enquanto tecnicamente não há problema em operar com 64 caracteres por linha, por exemplo, na prática, o diâmetro dos caracteres não passaria de três ou quatro pixels, considerando-se que o limite horizontal da tela é de 256 pontos (256/64 = 4). Os caracteres normais do micro têm uma largura de oito pixels, por onde se percebe que a imagem, num televisor comum, seria no mínimo sofrível.

O programa apresentado na listagem 2, cria uma tela de texto de 32 colunas por 24 linhas, ou 768 posições. Este formato é mais ou menos padrão para equi-

pamentos deste porte, e foi adotado em função da maioria das dificuldades de manipulação da tela de texto estarem relacionadas com o número de linhas disponíveis. Este formato permite ainda que seja mantido o desenho original dos caracteres do micro, já que apenas o espaçamento entre as linhas é modificado. O programa imprime letras maiúsculas e minúsculas em vídeo normal ou inverso, além de implementar opcionalmente novas funções, como repetição automática de teclas e feedback auditivo (beep).

COMO USAR O PROGRAMA

Digite cuidadosamente a listagem 2 e grave-a em fita. Em seguida, dê RUN; não havendo erro, a tela permanecerá cor de laranja durante, aproximadamente, 15 segundos. Prepare o gravador, dê ENTER e grave a versão final do programa. Para carregar e executar, digite a seguinte sequência em forma de comando direto:

CLEAR 2000:CLOADM:EXEC

O próprio programa se encarrega de reservar espaço no topo da RAM. Apenas 262 bytes, mais 637 bytes de tabela são usados; as rotinas de inicialização

são apagadas a fim de economizar memória. Os comandos CLS e PRINT e a tecla CLEAR funcionam normalmente, porém, a rotina não imprime caracteres gráficos. Para manter a compatibilidade com programas que utilizam esse recurso, é possível reverter à tela de texto normal digitando-se SCREEN 0. É possível ainda fixar a tela de texto original na cor de laranja; experimente comandar SCREEN 0,1. Para retornar à alta resolução, digite SCREEN 1.

A instrução COLOR, além de definir a cor que será usada por CLS e pela rotina de SCROLL, estabelece também a cor em que serão impressos os caracteres. Dessa forma, SCREEN 1,0: COLOR 0,1: CLS imprime caracteres em preto sobre fundo verde, enquanto SCREEN 1,0: COLOR 1,0: CLS imprime caracteres em verde sobre fundo preto (as instruções SCREEN e CLS, obviamente, não são obrigatórias). Já as rotinas de beep e auto-repeat podem ser ativadas da seguinte forma:

POKE 118,1 — ativa beep
POKE 118,0 — desativa beep
POKE 119,1 — ativa repeat
POKE 119,0 — desativa repeat

O valor armazenado nos endereços 250 e 251 é assumido como o intervalo

de tempo que uma tecla aguarda ao ser pressionada, antes de ser repetida. Este delay é ajustado inicialmente em 6500, mas pode ser modificado pokeando-se qualquer outro valor nestes endereços.

Da mesma maneira que SCREEN, é possível modificar também o PMODE em uso. Note, contudo, que não é seguro trocar o PMODE quando a impressão estiver situada na última linha da tela, pois a rotina de SCROLL perde os valores de referência e pode ocorrer perda de controle do cursor. Quanto à instrução RESET, ela não elimina o programa da memória mas faz o computador retornar à tela de texto normal. Para voltar à tela de alta resolução, digite SCREEN 1. Um RESET geral do sistema é obtido por POKE 113,0: EXEC 40999.

Embora seja curto, este programa executa várias operações, e uma análise dos procedimentos de programação envolvidos fica para outras edições de MICRO SISTEMAS. Por ora, vale constatar que com criatividade e um pouco de técnica, não há limitação que não possa ser superada, às vezes até com vantagens.

Claudio Costa é desenhista free-lancer e tem como hobby a programação de microcomputadores.

CONTABILIDADE ON-LINE PARA MICROCOMPUTADORES

A Contabilidade INTELSOFT é a única que realmente funciona em modo on-line. A qualquer momento você pode fazer consultas no vídeo para obter saldos de qualquer conta e lançamentos em formato de razão.

A Contabilidade INTELSOFT funciona tanto em micros de 8 bits, com sistema operacional CP/M, como em micros de 16 bits, compatíveis com o IBM PC. Todas as versões são funcionalmente idênticas podendo os mesmos arquivos serem usados por qualquer versão. Isso significa que ao mudar de computador você continua trabalhando da mesma maneira e aproveita integralmente o trabalho já feito.

Na tabela ao lado você encontra as características mais importantes da Contabilidade INTELSOFT. Use-a para facilitar a sua decisão de compra, comparando-a com as concorrentes.

PREÇO: 150 ORTNs para micros CP/M
200 ORTNs para micros MS-DOS
(compatíveis com IBM PC)

* Caso você queira experimentar o sistema antes de comprá-lo, a INTELSOFT fornece um pacote de demonstração por apenas 15 ORTNs. O pacote de demonstração é uma versão completa do sistema, sem nenhuma limitação, acompanhada do manual. O valor do pacote de demonstração pode ser aplicado integralmente, em ORTNs, na compra do sistema, em um prazo máximo de 60 dias.

compara você mesmo e nossa contabilidade com os concorrentes

CARACTERÍSTICAS	INTELSOFT	
Totalmente controlado por menus e realmente on-line.	SIM	
Número máximo de contas	65535	
Número máximo de lançamentos	65535	
Máximo de algarismos por valor	15	
Máximo de graus no plano	8	
Código de conta flexível e definido pelo usuário com número variável de dígitos por grau.	SIM	
Lançamentos de partida simples ou dobrada	SIM	
Históricos livres e padronizados	SIM	
Permite trabalhar com ou sem centavos	SIM	
Permite consolidar os dados de diversas empresas ou filiais	SIM	
Emite relatórios especiais para corretoras e distribuidoras de acordo com os modelos do Bencol Central	SIM	
Permite que os dados do sistema sejam convertidos para serem usados por pacotes como o dBASE II ou III, LOTUS 1-2-3, VISICALC ou por programas escritos em qualquer linguagem.	SIM	

Você possui um TK90X e quer desenvolver o seu próprio adventure?
Se a resposta for sim, este editor poderá auxiliá-lo nesta missão.

Editor de Adventures

Paolo Fabrizio Pugno

Antes de mais nada, vamos botar os pingos nos devidos lugares: os programas "Editor" e "Adventure" foram elaborados e adaptados para a linha ZX Spectrum a partir de uma idéia semelhante publicada na revista italiana "Microcomputer", para a linha Apple.

O assunto é adventure, este fascinante tipo de jogo que põe a dura prova nosso raciocínio, estratégia, paciência, e, às vezes, até mesmo sorte. Jogá-lo é uma coisa, elaborá-lo é outra totalmente diferente, onde devemos levar em consideração todas as possíveis situações que poderão ser enfrentadas pelo programa. Em resumo, não é nada fácil, embora não seja impossível. E tem mais, para cada adventure que escrevermos, teremos que bolar um programa gerenciador específico, que manipule os dados necessários. Isto faz com que horas e mais horas sejam perdidas na depuração de um maldito programa. Às vezes, só de pensar em enfrentar tais dificuldades, desistimos logo de saída. Mas, que tal um programa universal, isto é, que sirva para jogar todos os adventures que escrevermos? Tudo o que temos a fazer é "ensinar-lhe" as regras do jogo. Porém, é necessário que os dados lhe sejam fornecidos de uma maneira padronizada, e é aqui que entra em cena o "Editor".

Na realidade, são dois programas: com o "Editor" ensinamos as regras do jogo à máquina, que vai organizá-las de

uma forma que o "Adventure" possa manipulá-las. Os dados são salvos em fita, através da gravação de matrizes numéricas e alfanuméricas, que serão posteriormente carregadas pelo "Adventure". Desta forma, podemos inventar o adventure que quisermos, sem nos preocuparmos com o programa gerenciador. Como jogar um adventure feito por você mesmo perde logo a graça, a sugestão é realizar trocas entre amigos que possuam um ZX Spectrum ou TK90X; essa troca consistiria nas matrizes salvas em fita pelo "Editor".

EDITOR

Antes de rodar o "Editor", é obrigatório anotar tudo o que for necessário; fazer um mapa; dar números aos locais (começando por 1) e aos objetos; anotar os verbos e seus complementos; enfim, tudo o que depois será pedido pelo programa.

Dado o RUN, o "Editor" irá pedir o nome do jogo, o qual não deverá ter mais de dez caracteres, pois será utilizado como nome de uma matriz na hora de gravar os dados em fita. O programa então perguntará se o adventure é por pontos. Teclando S será pedido o escore a ser atingido para que o jogador alcance a vitória. Logo a seguir, devemos introduzir o número total de locais de jogo, após o que será pedida a descrição

de cada local. Essa descrição não deve ultrapassar três linhas (96 caracteres); o que, no entanto, pode ser alterado na linha 60, mas cuidado para não desperdiçar memória. Após introduzirmos cada descrição, devemos indicar que local é atingido quando nos movimentamos nas seis direções possíveis (Norte, Sul, Leste, Oeste, Suba e Desça). Introduzindo o zero para uma direção qualquer, iremos provocar o aparecimento da mensagem "não posso ir nessa direção", quando a mesma for executada nesse local; introduzindo o número do próprio local, não haverá deslocamento. Note-se que não é obrigatório nos locomovemos para locais vizinhos (no mapa). Aliás, quanto mais complicado for o mapa, mais interessante o adventure se tornará.

Na segunda parte do programa, é pedido o número de objetos existentes no jogo e, depois, o nome de cada um (sem utilizar artigo) e sua localização inicial. Aqui, há um truque: o local zero não existe no jogo, e serve para colocarmos temporariamente objetos que devem desaparecer ou para guardarmos outros que irão surgir posteriormente. Portanto, se uma chave só aparecer quando quebramos um vaso (por exemplo), seu local inicial será o zero e quando a ação "quebre o vaso" for executada, seu vetor posição será igual à posição do jogador. Não há limites de objetos por local. O nome do objeto deve ter, no máximo, 14 caracteres.

Agora vem a parte mais importante, os verbos: Pegue; Apanhe; Solte e Largue, para a manipulação de objetos (alguns verbos já estão programados no adventure por serem os mais utilizados, portanto, não é necessário defini-los).

Além dos verbos, podemos utilizar os seguintes comandos: Inventário (I), que mostra quais objetos estamos carregando; Pontos, informa sobre o escore; e Desisto, faz você recomeçar o jogo ou terminá-lo. E, é claro, Norte (N), Sul (S), Leste (L), Oeste (O), Suba e Desça, para os movimentos.

Para definir um verbo (e, portanto, uma ação), é necessário que tenhamos uma série de informações. Suponha que a descrição de um determinado local nos diz que há uma porta fechada ali. Estabelecemos que essa porta só pode ser aberta com uma chave, que deve estar com o jogador, e que ao abri-la, perdemos a chave e somos transportados para outro local qualquer. A ação será "abra a porta", e ela só pode ser executada nesse local onde há a porta e que, além disso, ao executá-la ganhamos alguns pontos.

Como codificar este exemplo? O micro vai perguntar:

1) Imperativo do verbo:

Abra.

2) Complemento:

Porta (sem utilizar artigo).

3) Resposta do micro:

"Ok. Puxa que lugar estranho!" (se quisermos que a resposta seja somente "Ok.", basta teclarmos ENTER).

4) Para que local sou transportado?

(Aqui deverá ser indicado o número do local de destino, sendo que 0 indicará a permanência no local).

5) Deslocamento de objetos:

Nesta pergunta, ao introduzirmos um número inteiro positivo, indicamos que o objeto que tem esse número irá aparecer no local. Um número inteiro negativo indica que o objeto que tem esse número (em módulo) irá desaparecer, ou melhor, sua posição será a sala zero, de que falamos há pouco. No nosso caso, se a chave for o objeto número 3, teclaremos: -3. Se o número indicado for zero, não haverá deslocamento de objetos.

6) Variação de escore (só se o jogo for por pontos):

Se o adventure for por pontos, introduziremos a variação no escore do jogador (positiva, negativa e nula) quando for executada esta ação.

7) Onde executar o comando?

Introduzindo um número inteiro positivo, indicamos que esta ação só pode ser executada no local que tenha este número. Um número inteiro negativo indica que esta ação só pode ser executada no local que tenha este número (em módulo). Zero indica que a ação

pode ser executada em qualquer lugar.

8) Objeto necessário:

Aqui, indicamos o número do objeto necessário para a execução desta ação. O objeto deve estar em poder do jogador para que esta ação seja executada. Zero indica que não é necessário objeto algum. No nosso caso, teclaremos 3 (o número da chave).

9) Mais de uma execução?

Se a ação puder ser executada várias vezes, teclaremos S, caso contrário, teclaremos N. Se teclarmos N, o micro somente executará esta ação uma vez, recusando-se a repeti-la. Por exemplo, não podemos quebrar o mesmo vaso duas vezes.

Note que se você quiser incluir o verbo "examine", deverá programá-lo para cada objeto que puder ser examinado. O programa permite que tenhamos diversos imperativos iguais, desde que os complementos sejam diferentes; assim como permite complementos iguais, se os imperativos forem diferentes. Então, podemos ter um mesmo adventure: abra porta, abra janela, abra garrafa, quebre garrafa, tampe garrafa, examine chave, examine vaso, examine garrafa etc., desde que forneçamos as informações adequadas para cada caso.

Observações importantes: o local -1 é o que indica que o jogador morreu ou foi eliminado do jogo. Portanto, se ao executar alguma ação o jogador levar a pior, devendo ser eliminado (por exemplo: salte despenhadeiro), basta transportá-lo para o local -1. Analogamente, o local -2 é o da vitória. Quando o adventure for resolvido, basta transportar o jogador para o local -2, que automaticamente o programa desvia para uma rotina de felicitações. A passagem para estes locais pode ser feita normalmente, seja através de consequência de uma ação (ou ações) como por deslocamento voluntário (N, S, L, O, Suba ou Desça).

Além disso, quando um objeto está na posição -1 significa que está em poder do jogador, o qual poderá carregar até cinco objetos. As posições dos objetos e do jogador não interferem entre si, pois são guardadas em variáveis distintas.

Por fim, será perguntado o local inicial de jogo, e, então, serão gravadas em fita as 14 matrizes que contêm os dados introduzidos. Devido às características do micro, é necessário que apertemos uma tecla qualquer antes do SAVE, portanto, cuidado para não se distrair. Se o VERIFY acusar erros, rebobine a fita e dê um GO TO 400.

Quanto à digitação, esta não apresenta maiores problemas; as escritas em INVERSE são obtidas teclando-se INVERSE VIDEO, antes do texto, e TRUE VIDEO, depois. Cuidado máximo nas li-



CP500/M⁸⁰, uma grande decisão para homens que tomam decisões.

O CP 500/M⁸⁰ agora também é compatível com o CP/M, o Sistema mais difundido no mundo. Isso significa que, somando o tradicional DOS 500 com o CP/M, você passa a dispor da maior biblioteca de programas aplicativos do mercado. E você poderá tirar vantagens de programas como Calc Star[®], dBase II[®], Word Star[®] e muitos outros, que vão agilizar tanto sua vida quanto a de sua empresa.

Para você, executivo, o CP 500/M⁸⁰ é uma poderosa ferramenta de auxílio a decisões gerenciais, com programas específicos em sua área de atuação, como análise de fluxo de caixa, controles de compras/custos/estoques e administração financeira. Como resultado final, sua empresa terá grandes ganhos de produtividade e eficiência. CP 500/M⁸⁰, um grande passo na informática com a Qualidade CP e a Tecnologia Prologica.

Características	Sistemas Operacionais Incluídos	
Memória RAM	SO-08 64 Kbytes	DOS 500 48 Kbytes
Memória ROM	2 Kbytes	16 Kbytes
Compatibilidade	CP/M [®]	TRS-DOS
Video	80 colunas por 24 linhas	64 ou 32 colunas por 16 linhas

Video 12" fósforo verde com controle de intensidade
Teclado profissional capacitivo com numérico reduzido
Interface para impressora paralela
Opera com 1 ou 2 drives de face simples ou dupla
Acesso a toda a Rede Nacional de Telemática, através de Interface serial padrão RS-232 C (opcional)

* Marcas Registradas

CP
COMPUTADORES PESSOAIS
TECNOLOGIA
PROLOGICA

nhas 410 e 430; elas devem ser iguais, exceto pela substituição dos SAVE pelos VERIFY. Uma dica é escrever a linha 410, editá-la e substituir o número de linha e os comandos SAVE por VERIFY.

ADVENTURE

Ao rodar o segundo programa, ele irá simplesmente perguntar o nome do adventure que desejamos carregar, e, logo após o término do carregamento, entrará automaticamente em execução.

Algumas recomendações: para movimentar-se, tecla "va para o norte" ou simplesmente N. Ao executar uma ação, você pode usar o artigo que quiser, tendo em mente que o programa reconhece somente a primeira e a última palavra da frase introduzida. Podemos então digitar: pegue martelo, pegue o martelo, pegue esse martelo ou pegue essa droga de martelo; que o efeito será o mesmo em todos os casos. Esse é o motivo pelo qual o Editor não aceita os artigos.

OUTROS MACETES

Se há uma sala escura que só revela seu conteúdo se acendermos uma vela

(por exemplo), então, é conveniente termos duas salas; sendo uma com a descrição da escuridão, sem outras saídas ou objetos; e outra, que é o seu equivalente, caso tenhamos luz. Definimos assim uma ação, "acenda vela", de tal forma que só poderá ser executada na sala escura e se tivermos a vela, e que além disso, nos transporte para a outra sala, com a outra descrição. E

um verbo, "apague vela", que faça o contrário.

Experimentalmente, conseguiu-se dimensionar matrizes referentes a 100 salas, 80 objetos e 150 verbos, o que acreditamos serem suficientes para a maioria dos casos. O comando POKE 23658,8 força o cursor para o modo C (maiúsculas). Se você quiser incrementar um pouco mais, experimente um POKE 23609,30.

UM EXEMPLO

Aqui está um exemplo de adventure com cinco salas. Tente rodá-lo. Os números estão na ordem em que o micro os pede.

Nome - "Teste"
É por pontos? - Não
Quanto locais? - 5

LOCAL 1

Descrição: "Estamos numa velha sala de estar. Há muito pó, aqui!"

NORTE	SUL	LESTE	OESTE	SUBA	DESCA
0	0	2	0	0	0

LOCAL 2

Descrição: "Estamos num corredor. Há uma escada que desce e uma porta".

NORTE	SUL	LESTE	OESTE	SUBA	DESCA
0	0	0	1	0	4

LOCAL 3

Descrição: "Que lugar estranho! Há um botão na parede".

NORTE	SUL	LESTE	OESTE	SUBA	DESCA
0	0	0	2	0	0

LOCAL 4

Descrição: "Está muito escuro, mas percebo uma tocha apagada na parede".

NORTE	SUL	LESTE	OESTE	SUBA	DESCA
0	0	0	0	2	0

Adventure

```
1) REM *****
2) REM ** ADVENTURE **
3) REM *****
4) REM
10 POKE 23658,8: BORDER 1: PAPER 5: INPUT "QUAL O NOME DO ADVENTURE QUE VOCE QUER JOGAR?": A$=CLS
15 LOAD A$ DATA 1: LOAD "LOCAL 1": DATA L$(): LOAD "DESCR." DATA M$(): LOAD "OBJETOS" DATA O$(): LOAD "POS.OBJ." DATA P$(): LOAD "VERBOS" DATA V$(): LOAD "COMP.L." DATA C$(): LOAD "MENSAGENS" DATA H$(): LOAD "VOZPARA" DATA D$(): LOAD "DES.OBJ." DATA A$(): LOAD "P.OBJ." DATA U$(): LOAD "OBJ.NEDES." DATA B$(): LOAD "REP.VER." DATA K$()
20 CLS: LET SCORE=0: LET POS=Z(1): LET SPON=Z(2): LET PONMAX=Z(3): LET NL=Z(4): LET NO=Z(5): LET NU=Z(6): DIM E(NU)
25 LET J=0: LET POSI=POSI: DIM Q(NU): FOR I=1 TO NO: LET Q(I)=P(I): NEXT I: CLS
30 PRINT L$(POS):
35 IF POS=1 THEN GOTO 470
40 IF POS=2 THEN GOTO 495
45 LET QW=0
50 FOR I=1 TO NO
55 IF P(I)<>POS THEN GOTO 75
60 IF QW THEN GOTO 70
65 PRINT "OS OBJETOS VISIVEIS SAO:" LET QW=1
70 PRINT Q$1, 10 14,
75 NEXT I
80 INPUT FLASH 1: "O QUE FAZEMOS AQUI?": H$=PRINT "H$":
85 IF H$="" THEN CLS: GOTO 30
90 LET A$="": FOR F=1 TO LEN H$
95 IF H$(F)="" THEN GOTO 105
100 LET A$=A$+H$(F): NEXT F
105 LET B$="": FOR I=1 TO NU
110 IF H$(I)="" THEN GOTO 120
115 LET B$=B$+H$(I): NEXT I
120 LET QW=0: FOR I=1 TO NU
125 IF A$(I)VS(I), TO CODE VS(I),
```

```
5) THEN GOTO 195
130 IF B$(VS(I), TO CODE VS(I),
5) THEN GOTO 195
135 IF Y(I) AND E(I) THEN PRINT FLASH 1: "Mas ja fiz isso!": GOTO 190
140 IF B(I) THEN IF P(I)<>1 THEN PRINT "Ajuda nao posso fazer isso!": GOTO 190
145 IF U(I)=0 THEN GOTO 160
150 IF SPON U(I)=1 THEN IF POS=ABS U(I) THEN PRINT "Nao posso fazer isso aqui!": GOTO 190
155 IF U(I)=0 AND U(I)=POS THEN PRINT "Nao posso fazer isso aqui!": GOTO 190
160 PRINT "H$":
165 IF D(I) THEN LET POS=D(I): IF POS=1 THEN PRINT "L$1: POSI":
170 IF F(I)=0 THEN GOTO 185
175 IF SPON F(I)=1 THEN LET P(I)=POS: GOTO 185
180 LET P(ABS F(I))=0
185 LET E(I)=1: IF SPON THEN LET SCORE=SCORE+A(I)
190 LET J=NU: LET QW=1
195 NEXT I
200 IF SCORE=PONMAX THEN GOTO 495
205 IF QW THEN GOTO 35
210 IF NOT A$="PONTOS" THEN GOT 0 220
215 PRINT "TEMOS ATE AGORA": SCORE: "PONTOS": GOTO 80
220 IF NOT A$="PEGUE" OR A$="A PARHE" THEN GOTO 280
225 IF B$="" THEN PRINT "DIGA-ME PELO MENOS O QUE": GOTO 80
230 IF J=5 THEN PRINT "NAO POSSO CARREGAR MAIS NADA": GOTO 80
235 LET QW=0: FOR I=1 TO NO
240 IF NOT (B$(VS(I), TO CODE VS(I), 1, 1) AND P(I)=POS) THEN GOTO 245
245 PRINT "OK, "A$": "I": "B$":
250 LET QW=1
255 LET P(I)=1
260 LET J=J+1
265 NEXT I
```

```
270 IF NOT QW THEN PRINT "NUNCA FIZEMOS NADA!": GOTO 80
275 GOTO 80
280 IF NOT A$="SOLIE" OR A$="LARGUE" THEN GOTO 335
285 LET QW=0
290 FOR I=1 TO NO
295 IF NOT (B$(VS(I), TO CODE VS(I), 1, 1) AND P(I)=1) THEN GOTO 3
300 PRINT "OK, "A$": "I": "B$":
305 LET QW=1
310 LET P(I)=POS
315 LET J=J+1
320 NEXT I
325 IF QW THEN GOTO 35
330 PRINT "NAO TENHO": "B$": CONOSCO": GOTO 80
335 IF A$="DESISTO" THEN GOTO 345
340 CLS: PRINT "NUNCA ADVENTURE VOCE FEZ": SCORE: "PONTOS, ISTO E": SCORE/PONMAX*100: "Z": "DO TOTAL": GOTO 475
345 IF NOT (A$="I" OR A$="INVENTARIO") THEN GOTO 400
350 LET QW=0
355 FOR I=1 TO NO
360 IF P(I)<>1 THEN GOTO 385
365 IF QW THEN GOTO 380
370 PRINT "ESTOU CARREGANDO":
375 LET QW=1
380 PRINT IAB 10: BRIGHT 1: 0%: 1 TO 14):
385 NEXT I
390 IF QW THEN GOTO 80
395 PRINT "NAO ESTOU CARREGANDO NADA": GOTO 80
400 IF NOT ((A$="N" AND H(POS,1)) OR (A$="VA" AND B$="NORTE" AND H(POS,1))) THEN GOTO 410
405 LET POS=H(POS,1): GOTO 30
410 IF NOT ((A$="S" AND H(POS,2)) OR (A$="VA" AND B$="SUL" AND H(POS,2))) THEN GOTO 420
415 LET POS=H(POS,2): GOTO 30
420 IF NOT ((A$="L" AND H(POS,3)) OR (A$="VA" AND B$="LESTE" AND H(POS,3))) THEN GOTO 430
425 LET POS=H(POS,3): GOTO 30
```

```
430 IF NOT (A$="E" AND H(POS,4)) OR (A$="VA" AND B$="OESTE" AND H(POS,4)) THEN GOTO 440
435 LET POS=H(POS,4): GOTO 30
440 IF NOT (A$="SUBA" AND H(POS,5)) OR (A$="VA" AND B$="SUBA" AND H(POS,5)) THEN GOTO 450
445 LET POS=H(POS,5): GOTO 30
450 IF NOT (A$="DESCA" AND H(POS,6)) OR (A$="VA" AND B$="DESCA" AND H(POS,6)) THEN GOTO 460
455 LET POS=H(POS,6): GOTO 30
460 IF POS=1 THEN PRINT "L$1: POSI":
465 PRINT FLASH 1: "OK, A SUA ORDEM": GOTO 80
470 PAUSE 5: BORDER 0: PAPER 0: INK 0: CLS: PRINT AT 10,0: FLASH 1: BRIGHT 1: "SINIO MUITO, MAS VOCE PERDEU...": "FINISH SORTE NA PROXIMA VEZ...": "FIM DO JOGO":
475 INPUT "OK?": Y$=PRINT "Y$":
480 IF Y$="" THEN STOP
485 FOR I=1 TO NO: LET P(I)=0:
490 IF I=1 THEN LET POS=POSI: LET SCORE=0: FOR I=1 TO NU: LET E(I)=0: NEXT I: LET J=0
495 BORDER 1: PAPER 5: INK 0: CLS: GOTO 30
495 PAUSE 5: BORDER 4: PAPER 0: INK 1: CLS: PRINT AT 10,0: FLASH 1: "MEUS PARABENS! VOCE CONSIGUIU COMPLETAR O ADVENTURE!":
FOR F=0 TO 40: SOUND 1,0: NEXT F: IF SPON THEN PRINT "VOCE FEZ": SCORE: "PONTOS, DE UM TOTAL DE": PONMAX: "PONTOS": "ISTO E": SCORE/PONMAX*100: "Z DO TOTAL": GOTO 475
```

BRASIL TRADE CENTER

Comércio, Indústria e Participações S.A.

COMPUTER SHOPPING

Mesas para Microcomputadores

Fabricação própria • Cores discretas • Desenho moderno



EQUIPAMENTOS

Micros das linhas: Sinclair • TRS-80 • CCE • Unitron • Dismac • TK 2000 II • TK 85 • Exato Pro • Impressoras Modem • Etc.

SUPRIMENTOS

Periféricos • Disketes • Fitos • Drives • Placas de Expansão • Interfaces • Formulário Continuo • Cabos • Arquivos para Disketes • Etc.

SOFTWARE

Programas e Jogos variados para todas as linhas e marcas de Computadores

VIDEOS

Curso de Inglês em vídeo cassete • Telão • Suporte para TV • Transcodificação para todos os sistemas • Fitos: VHS - BETA-U-MATIC e para limpeza de cabeça • Baterias para 2 e B hs. • Iluminadores • Cabos de extensão para Câmaras • Bolsas para Câmaras e Vídeos

Nas lojas Brasil Trade Center, você também encontra vídeo-game, cartucho com jogos, som, telefonia, etc., das melhores marcas.

VISITE UMA DE NOSSAS LOJAS E COMPROVE O QUE ANUNCIAMOS

Matriz: Av. Epitácio Pessoa, 280
Ipanema-RJ — Tels.: 259-1299/259-1499

Filiais: Rua da Assembléia, 10 — Loja S-112
Centro-RJ — Tel.: 222-5343

Rua Silva Vale, 416 — Cavalcante-RJ
Tels.: 592-3047/592-3098

Rua Lopes Trovão, 134/Sala 201
Niterói-RJ — Tel.: 710-3659

Rua Conde de Bontim, 229 — Loja A
Tijuca-RJ — Tel.: 284-2031

SERVIÇO EXPRESSO REMETEMOS PARA TODO O BRASIL

OFERTA VÁLIDA COM A APRESENTAÇÃO DESTE ANÚNCIO



Gilson Roberto Viana

Thunderbird

Thunderbird é um programa animado, baseado numa batalha aérea onde você tem, como árdua tarefa, a missão de pilotar um avião de bombardeio por uma região montanhosa e infestada de mísseis antiaéreos e caças inimigos. Cada projétil inimigo abatido por um tiro ou uma bomba aumentará sua pontuação em 10 pontos.

Cada vez que você colidir com um obstáculo (montanha, míssil ou caça), perderá uma nave e o jogo só se reiniciará quando for pressionada a tecla **NEWLINE**. O jogo acaba quando você perder suas três naves.

A cada reinício de partida, o leitor pode modificar a velocidade de acordo com o nível escolhido, que pode ser lento (1), médio (2) e rápido (3). Entretanto, se você estiver realizando uma competição, é aconselhável manter o mesmo nível, para que todos os jogadores pilotem com o mesmo grau de dificuldade.

Caso você consiga quebrar o recorde antigo, terá o privilégio de escrever seu nome no quadro dos campeões. O nome pode ter, no máximo, sete letras e, se for maior, deve ser completado com pontos.

Os controles podem ser usados simultaneamente. São eles:

- 1) Digite a listagem 1;
- 2) Digite **GOTO 100** e **NEWLINE**;
- 3) Quando for pedido endereço inicial, digite **16514** e **NEWLINE**;
- 4) Digite os códigos em linguagem de máquina da listagem 2 e, quando acabar digite "...";
- 5) Quando, novamente for pedido endereço inicial, digite **30720** e **NEWLINE**;
- 6) Digite os códigos em linguagem de máquina da listagem 3 e, quando terminar, digite "XX";
- 7) Apague (delete) as linhas de **100** até **210**;
- 8) Finalmente, se tudo estiver correto, prepare e fita para gravar e digite **RUN** e **NEWLINE**.

Figura 1

6 — desce; 7 — sobe; 9 — dispara bombas (máximo de três bombas) e 0 — dispara tiros (máximo de cinco tiros).

Veja, na figura 1, como você deve proceder na digitação do programa.

Gilson Roberto Viana é estudante do Curso de Engenharia Elétrica, na UFPR, e Programador autodidata há dois anos.

```
1 REM 11111111111111111111
111111111
10 DIM K$(1110)
20 RAND USR 16528
30 SAVE "THUNDERBIRD"
40 RAND USR 16535
100 CLS
105 PRINT "ENDEREÇO INICIAL ?"
107 INPUT N
110 PRINT N
115 LET B=N
120 LET A$=""
125 IF A$(2 TO 7)="" THEN INPUT A$
130 IF A$(2 TO 7)="" THEN GOTO 125
135 GOTO (A$(1)-1)*(-40)+(A$(2)-1)*(-50)+140
140 POKE N,(CODE A$(2)-28)*16+CODE A$(2)-28
```

Listagem 1

```
16514 2A 10 40 11 00 00 19 11
16522 00 78 01 55 04 C9 CD 82
16530 40 E8 ED 80 C9 CD 82 40
16538 ED 80 C3 88 78 00 00 00
```

Listagem 2

```
30720 80 80 80 80 80 80 80 97
30728 97 80 89 AD 8A 83 A9 AA
30736 87 A7 AE B7 A9 80 97 97
30744 80 80 80 80 80 80 80 80
30752 80 00 2C 2E 31 38 34 33
30760 00 38 2E 26 33 26 00 16
30768 00 32 2E 28 37 34 00 38
30776 2E 38 39 2A 32 26 38 80
30784 07 03 84 82 A6 8D AE B2
30792 84 80 AF 84 AC A6 A9 84
30800 87 80 80 85 81 A6 A8 A6
30808 87 80 89 AA B2 85 84 80
30816 05 1F 85 1C 1C 1C 1C 1C
30824 1C 00 0F 0F 0F 0F 0F 0F
30832 0F 00 80 1C 1C 1C 1C 1C
30840 1C 80 1C 1E 25 25 25 80
30848 01 01 58 41 54 41 A3 08
30856 03 08 08 03 03 03 08 03
30864 08 03 08 03 08 03 08 03
30872 E5 D5 C5 2A 0C 40 11 EE
30880 02 19 5F 7E FE 25 20 05
```

```
30888 36 1C 28 18 F6 FE 80 28
30896 01 34 78 C1 D1 E1 C9 2A
30904 0C 40 01 F5 02 09 3E 1C
30912 0E 20 04 28 0E 28 FC 3E
30920 80 8E 28 09 35 23 8E C8
30928 36 25 18 F9 E1 2A 84 78
30936 16 08 E5 06 12 7E FE 76
30944 28 06 C6 80 77 23 18 F5
30952 23 10 F2 CD FD 7A E1 15
30960 20 E8 C9 AF D8 FE C8 47
30968 C0 21 80 78 7E FE 05 C8
30976 34 2A 82 78 23 97 BE 20
30984 03 36 16 C9 C8 66 28 04
30992 77 CD 98 78 C3 85 79 AF
31000 D8 FE C8 4F C0 21 81 78
31008 7E FE 03 C8 34 2A 82 78
31016 23 11 21 00 19 3E 87 BE
31024 CA 8C 79 C3 4C 79 2A 82
31032 78 23 11 21 00 19 7E FE
31040 8A C8 FE 87 20 F7 36 00
31048 19 8E 28 FC 97 8E 20 03
31056 36 87 C9 C8 46 28 04 77
31064 CD 98 78 C3 8C 79 2A 84
31072 78 23 7E FE 8A C8 FE 16
31080 20 F7 36 00 23 BE 28 FC
31088 97 8E 20 04 36 16 18 E9
31096 CD 92 79 C8 46 28 E2 77
31104 CD 98 78 18 DC FE 16 28
31112 09 FE 87 C0 E5 21 81 78
31120 18 04 E5 21 80 78 35 E1
31128 C9 21 86 78 34 7E 21 00
31136 09 23 3D 20 FC 23 3E 55
31144 86 4F 3E 0E A6 C8 5F 28
31152 F4 2A 0C 40 11 92 02 19
31160 11 21 00 36 08 ED 52 3D
31168 20 F9 79 FE 7F 28 06 FE
31176 FF C0 36 13 C9 36 2D C9
31184 23 8E 28 28 02 36 00 28
31192 8E 20 09 23 23 8E CA 78
31200 7A 28 18 0C 7E FE 82 CA
```

```
31208 D4 78 CD 85 79 36 08 23
31216 C3 9C 7A 36 00 E5 09 7E
31224 CD 26 7A E1 28 FE 08 28
31232 03 AF ED 42 7E FE 03 28
31240 06 FE 00 20 06 36 2D 23
31248 C3 81 7A FE 86 28 02 FE
31256 82 CA D4 78 36 00 CD 98
31264 78 CD 85 79 18 E9 D5 C5
31272 5F 23 16 07 03 AF ED 42
31280 7E FE 86 28 04 15 20 F5
31288 78 C1 D1 C9 36 00 1A FE
31296 08 28 02 09 09 AF ED 42
31304 7E FE 00 20 03 12 18 F5
31312 FE 03 20 03 12 18 EC 28
31320 7E FE 00 C2 13 7A 36 13
31328 13 23 C3 9C 7A 2A 84 78
31336 23 11 87 78 01 21 00 E5
31344 3E 0C 36 00 09 3D 20 FA
31352 E1 23 7E FE 00 CA 9F 7A
31360 E5 7E FE 8A 20 03 E1 18
31368 F0 FE 00 28 0F FE 08 CA
31376 D0 79 FE 2D CA F3 79 FE
31384 13 CA 3C 7A 09 18 E2 CD
31392 99 79 CD 5E 79 CD 36 79
31400 18 11 2A 0C 40 11 64 00
31408 19 22 84 78 11 07 00 19
31416 97 18 27 2A 82 78 E5 D1
31424 01 21 00 AF D8 FE C8 5F
31432 28 06 C8 67 20 20 09 09
31440 ED 42 7E FE 00 28 07 FE
31448 03 28 13 C3 D5 78 97 12
31456 18 12 36 82 28 8E 20 F3
31464 36 86 23 22 82 78 CD F3
31472 78 CD 17 79 CD 87 78 CD
31480 FD 7A C3 65 7A 01 01 0F
31488 08 79 80 20 F8 C9 2A 0C
31496 40 11 DA 02 19 E8 21 10
31504 00 19 C9 CD 06 78 13 23
31512 1A FE 00 C8 8E 28 F7 96
31520 D0 CD 06 78 01 06 00 ED
```

Listagem 3

COMPUTAÇÃO É COM A CAMPUS

BEST SELLERS:

LANÇAMENTOS

JANEIRO/86

- 1 MUMPS — Lobo, M. C. — Cr\$ 49.000
- 2 Inteligência Artificial em BASIC — James, M. — Cr\$ 90.000
- 3 VISITRENO/VISILOT: Guia do Usuário Brasileiro — Christmann, R. U. — Cr\$ 54.000
- 4 LOGO: Uma Introdução — Goodyear, P. M. — Cr\$ 82.000
- 5 Linguagens de Programação para Micros — Marshall, G. — Cr\$ 72.000
- 6 Pascal para Micros — James, M. — Cr\$ 68.000
- 7 Técnica de Gerenciamento de Arquivos — Claybrook, B. G. — Cr\$ 125.000

APPLE

- 8 Como Programar seu APPLE — Cook, R. A. Hartnall, T. — Cr\$ 41.000
- 9 Jogos Gráficos para o APPLE — Colatta, P. — Cr\$ 82.000
- 10 Manual APPLE em BASIC — Kanteris, N. — Cr\$ 52.000

BASIC

- 11 BASIC para Aplicações Comerciais — Hergert, O. — Cr\$ 55.000
- 12 BASIC com Estilo — Nagin, P. e Ledgard, H. F. — Cr\$ 48.000
- 13 BASIC Básico — Pereira F9, J. C. — (5ª edição revisada e ampliada) — Cr\$ 64.000

LINGUAGEM DE MÁQUINA

- 14 Além do BASIC — Linguagem ASSEMBLY para a Linha SINCLAIR — Santos, N. — Cr\$ 60.000

- 15 BASIC Rápido: Além do BASIC TRS-80 — Gratzler, E. A. e Gratzler, T. G. — Cr\$ 99.000
- 16 Manual do Microprocessador Z-80 — Barden Jr., W. — Cr\$ 109.000
- 17 Programação em ASSEMBLER a Linguagem da Máquina (2ª edição) — Alaxender, O. C. — Cr\$ 68.000
- 18 Desafio: Os Mais Excitantes Jogos em BASIC — Botelho, A. J. L. — Cr\$ 87.000
- 19 Primeiros Passos com o seu SINCLAIR (TKs) — Hughes, C. — Cr\$ 49.000
- 20 Programas Administrativos em BASIC SINCLAIR — Karsten, L. — Cr\$ 57.000
- 21 30 Programas em BASIC para Computadores Pessoais — Chance, O. — Cr\$ 92.000
- 22 1001 Aplicações para seu Computador Pessoal — Sawusch, M. — Cr\$ 88.000

ESTA É APENAS UMA SELEÇÃO DE NOSSOS TÍTULOS. PARA MAIORES INFORMAÇÕES SOLICITE NOSSO CATÁLOGO.

Editora Campus

Rua Barão de Itapagipe 55 - 20261 - RJ - RJ - Tel.: (021) 284 8443

SIM DESEJO RECEBER AS PUBLICAÇÕES REFERENTES AOS NÚMEROS ASSINALADOS:

1	2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21	22

Nome:
Endereço:
Cidade: CEP: Estado:

- ☐ Mediante envio de cheque nominal à Editora Campus Ltda. (O porta é por nossa conta e sua encomenda chega mais rápido.)
Cheque nº:
Banco:
- ☐ Mediante pagamento contra recebimento dos livros pelo reembolso postal, acrescido de despesas postais.

Importante: Preços válidos até 28.02.86. Após esta data informar se concorda com o envio do pedido com reajuste.
☐ SIM ☐ NÃO

NOSSAS PUBLICAÇÕES ESTÃO A VENDA TAMBÉM NAS BOAS LIVRARIAS

REPRESENTANTES AUTORIZADOS

Para sua melhor comodidade, a ATI Editora Ltda. coloca à sua disposição os seguintes endereços de seus representantes autorizados

BELO HORIZONTE
Maria Fernanda G. Andrade
Caixa Postal 1687
Tel.: (031) 335-6645



RIO DE JANEIRO
ATI Editore Ltda.
Av. Pres. Wilson, 165 - gr. 1210
CEP 20030 - Tel.: (021) 262-6306

SÃO PAULO
Rua Oliveira Oias, 153
Jardim Paulista
CEP 01433 - Tel.: (011) 853-3574

PORTO ALEGRE
Com. Rep. Odilon Ltda.
Rua Vol. da Pátria, 323 - Cj. 309
CEP 90000 - Tel.: (0512) 24-8200 R. 309

**Micro
Sistemas**

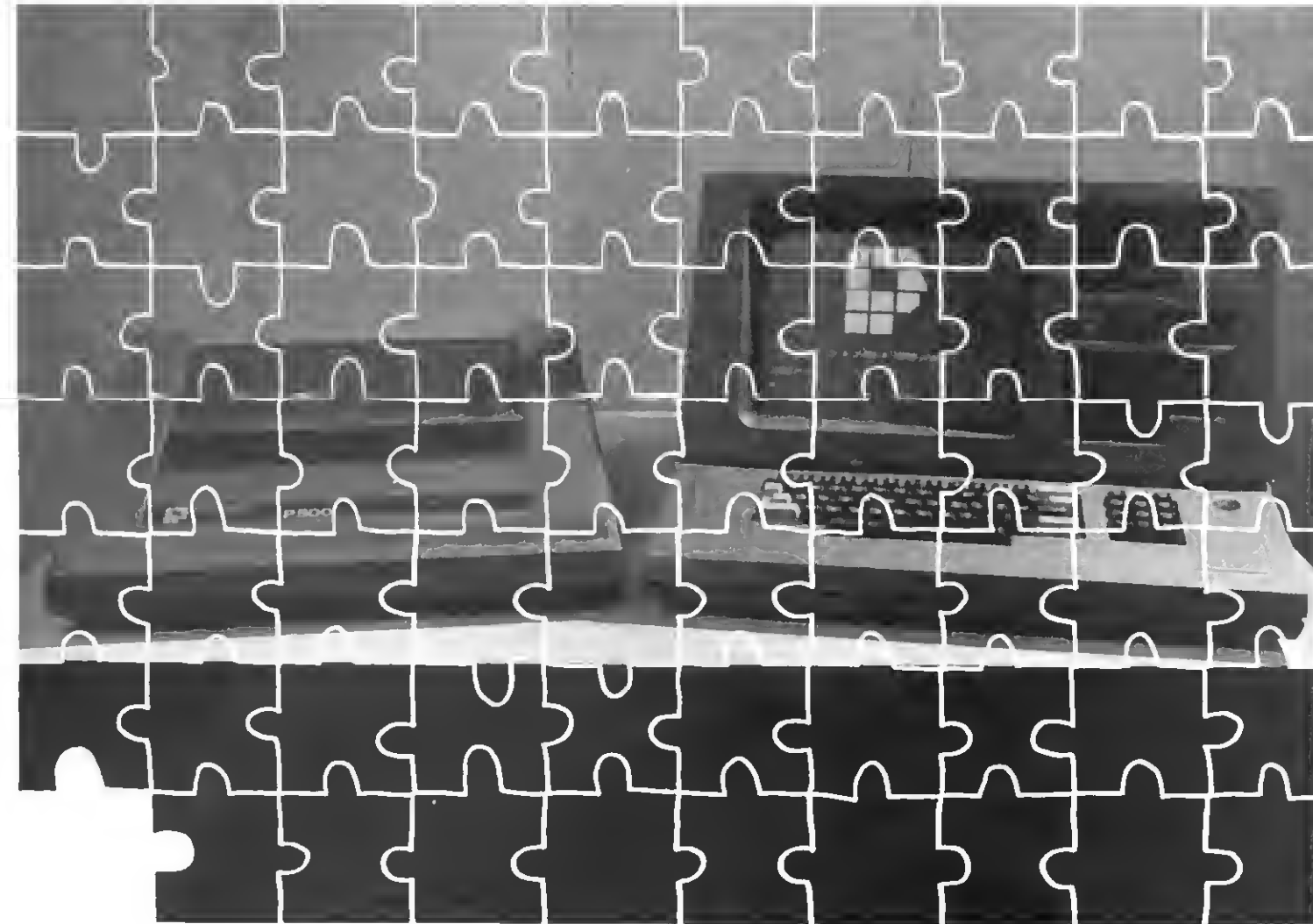

```

3150 RETURN
3160 PRINT"? ";
3170 RETURN
3180 PRINT"N5 ";
3190 RETURN
3200 PRINT"L0 ";
3210 RETURN
3220 GOT03160
3230 PRINT"PA ";
3240 RETURN
3250 '
3260 PRINT@192,"que esta' com fome";
3270 GOSUB3810
3280 PRINT@320,"que voce de um pouco de Diamantes";
3340 GOSUB@920
3350 RETURN
3360 '
3370 PRINT@192,"que nso esta' com fome";
3400 GOSUB3810
3410 PRINT@320,"que sia da um pouco da Diamantes";
3480 GOSUB3920
3490 RETURN
3500 GOSUB480
3510 H=I+O-9:H=8
3520 B=4+E-5:I=14
3530 C=0:P=R=0
3540 GOT01010
3550 '
3560 PRINT@192,"que nso esta' com fome";
3570 GOSUB3810
3580 PRINT@320,"qua ele de um pouco de Diamantes";
3640 GOSUB3920
3670 RETURN
3680 '
3690 PRINT@192,"que nso esta' com fome";
3700 GOSUB3810
3710 PRINT@320,"que sie de um pouco de Diamantes";
3770 GOSUB3920
3780 RETURN
3810 '
3820 PRINT@256,"Voce nao poders psssr sts ";
3830 RETURN
3840 '
3850 RETURN
3860 "ROTINA ALEATORIA
3900 '
3910 RETURN
3920 '
3930 BA=RM0(400/L1)+25
3940 G=B+G4
3950 PRINT@384,"E voce ganhou ";IG4;
3960 PRINT@448,"Diamntes ds Orion.";"
3970 A(C,O,L1)=1
3980 CA=CA+i:IFK=1THENRETURN
3990 IFCA=K4THEN4110
4000 IFL1=1THEN4238
4010 RETURN
4020 '
4060 G4=RD(350/L1)
4070 G=G-G4
4080 PRINT@384,"E vocs perdeu ";IG4;
4090 PRINT@448,"Diamntes de Orion.";"
4100 RETURN
4110 '
4120 K=1
4130 PRINT@576,"Voce schou s Estrelas do Poder";
4140 GOSUB30000
4150 RETURN
4160 D0SUB440:=G0SUB660:=PRINT@0,"Voce importouno um Verme do";PRINT@64,"
Pavor, nests quadrante, sie coma";
4170 '
4180 PRINT@128,"Diamntes de Orion e diz...";
4190 '
4200 RETURN
4210 D0SUB4120
4220 DOT01200
4230 IFCA=CBTHEN4110
4240 RETURN
4250 G=0-500:IFD@THENG=0
4260 PRINT@384,"que perdsu muitos ";
4270 PRINT@448,"ds ssus Osmntes, Rastarsa";
4280 PRINT@512,"ths "IG;" Oismntes ds Orion";
4285 GOSUB40000
4290 RETURN
4300 PRINT@384,"Voce recsbu Diamntes de ";ICA;
4301 PRINT@448,"Verms do Pavor...am ";MI;
4310 PRINT@512,"Jagads.";
4320 RETURN
4330 PRINT@256,"Os asteroidas se fechsa atrs";PRINT@320,"da voce";
4340 '
4350 PRINT@0,"Voce stravesse uns passasse";PRINT@64,"entre os asteroide
s";
4360 RETURN
6000 AA=98:BB=16:EE=14
6010 FORJJ=1TOEE
6020 XX=AA-JJ+JJ:YY=BB+JJ:SET(XX-1,YY):SET(XX,YY)
6030 YY=BB-JJ:SET(XX,YY):SET(XX-1,YY)
6040 YY=BB:SET(XX,YY):SET(XX-1,YY)
6050 XX=AA:SET(XX,YY):SET(XX-1,YY)
6060 YY=BB+JJ:SET(XX,YY):SET(XX-1,YY)
6070 YY=BB-JJ:SET(XX-1,YY):SET(XX,YY)
6080 XX=AA-JJ-JJ:SET(XX-1,YY):SET(XX,YY)
6090 YY=BB:SET(XX-1,YY):SET(XX,YY)
6100 YY=BB+JJ:SET(XX-1,YY):SET(XX,YY)
6110 NEXTJJ
6120 RETURN
6500 KK=STRING$(28,153):FORZ3=@T0B:PRINT@99+64-Z3,KY" 'NEXTZ3
6510 RETURN
6600 PRINT@168,CHR$(171)+CHR$(172)+STRING$(12,128)+CHR$(168)+CHR$(174)+CH
R$(129);
6610 PRINT@232,CHR$(130)+CHR$(139)+CHR$(164)+CHR$(144)+CHR$(160)+STRING$
(7,176)+CHR$(128)+CHR$(176)+CHR$(142)+CHR$(131);
6620 PRINT@299,CHR$(170)+CHR$(131)+CHR$(128)+STRING$(4,176)+CHR$(144)+CH
R$(130)+CHR$(171);
6630 PRINT @363,CHR$(149)+CHR$(128)+CHR$(128)+CHR$(141)+CHR$(136)+CHR$
(141)+CHR$(136)+CHR$(133)+CHR$(128)+CHR$(128)+CHR$(149);
6640 PRINT@426,CHR$(154)+CHR$(128)+CHR$(160)+STRING$(7,176)+CHR$(128)+CH
R$(138)+CHR$(144);
6650 PRINT@489,CHR$(160)+CHR$(133)+CHR$(128)+CHR$(129)+CHR$(129)+CHR$(12
0)+CHR$(128)+CHR$(128)+CHR$(138)+CHR$(128)+CHR$(128)+CHR$(129)+CHR$(129)

```

[illegible]

SOLUÇÕES NA MEDIDA EXATA DE SUAS NECESSIDADES.



A Filcres tem sempre em estoque e para pronta entrega tudo o que você precisa para o seu CPD: fitas impressoras, formulários, diskettes, e mais uma infinidade de outros acessórios. E conta ainda com uma equipe de profissionais altamente qualificados, prontos a lhe prestar o melhor atendimento, Tudo isso com a qualidade que v. exige e um preço sempre compatível com seu orçamento. É só telefonar e esquecer seu problema. A Filcres leva o suprimento que falta até você.

20 Anos
de
Sucesso



AJUDANDO A DESENVOLVER TECNOLOGIA

Rua Aurora, 165 - SP - Tel. PBX (011) 223-7388 - Telex (011) 31298 Direto - SP - 223-1446/
222-3458/220-9113/220-7954 Direto outros Estados: 222-5430/221-0326/223-7649/222-0284



Batalha Naval

Renato da Silva Oliveira e Pierluigi Piazzi

O programa que apresentamos a seguir, ilustra como uma idéia simples pode dar origem a um jogo de razoável complexidade. Uma das coisas mais triviais que se pode pensar, ao fazer um jogo, é num atirador e num alvo. Obviamente, nesses casos, o usuário costuma fazer o papel do atirador. Em nosso exemplo, usamos um navio lançando cargas em profundidade num submarino.

Um pouco mais difícil é engendrar uma forma de fazer com que o usuário tenha que pensar antes de dar seu tiro. Em nosso jogo é praticamente inevitável o uso do fosfato cerebral! O usuário é obrigado a pensar para obter êxito (e isso, para muitos, dói!).

A idéia por trás de tudo é bastante simples. Para atingir o submarino, o usuário deve dar a posição em que a carga será lançada e isso é feito através da introdução de três números: um que indique a posição na direção Norte-Sul, outro que indique a posição na direção Leste-Oeste e um terceiro indicando a profundidade em que a carga deverá explodir.

O submarino é colocado, ao acaso, dentro de uma região cúbica de 8 Km na direção Norte-Sul, 8 Km na direção Leste-Oeste e 4 Km de profundidade. Nessa mesma região, está colocado também o navio lançador de cargas. Observe a figura 1.

Note que o navio sempre fica no vértice do paralelepípedo formado pela região de 8x8x4 quilômetros cúbicos e que o submarino é sempre colocado numa posição de coordenadas inteiras. Por exemplo, ele nunca vai estar em uma profundidade de 2,5 Km (estará a 2 ou 3 Km).

Ao ser executado, o programa pede a introdução dos três números indicadores

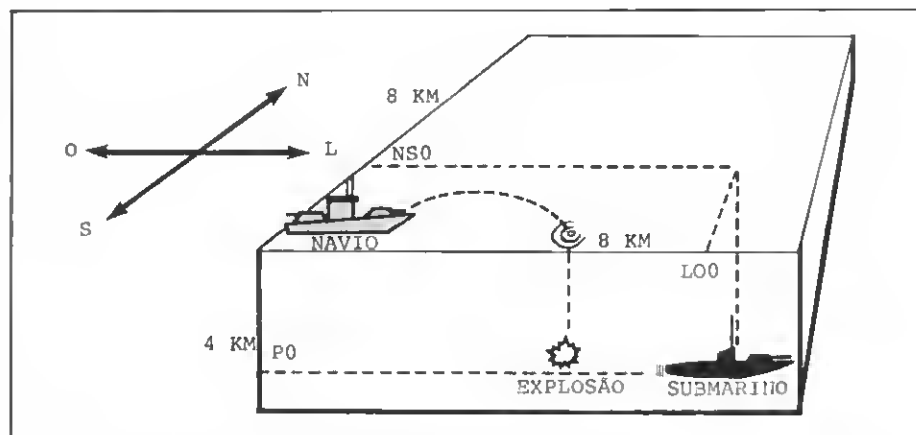


Figura 1 - O navio e o submarino estão numa região de 8x8x4 Km³.

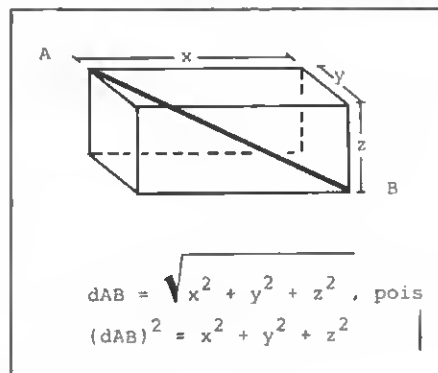


Figura 2

da posição. Inicialmente, a distância ao longo da direção Norte-Sul (N/S), depois a distância ao longo da direção Leste-Oeste (L/O) e, finalmente, a profundidade (P). O primeiro tiro deve ser dado ao acaso. Lembre-se que os valores para N/S e L/O devem estar entre 0 e 8, e os valores para P devem estar entre 0 e 4. Caso esses limites sejam desrespeitados, o tiro será perdido.

Como resposta a cada tiro, o programa dá uma dica ao usuário: a distância a que a carga explodiu do submarino. Essa

distância é calculada através do "Teorema de Pitágoras" aplicado a um espaço tridimensional (Teorema de "Tritágoras"). A ilustração do teorema pode ser vista na figura 2.

É a partir dessa informação que os tiros subsequentes devem ser dados. Por exemplo, imagine que seu primeiro tiro tenha sido [4] [1] [3] e que a distância obtida como dica tenha sido 4. Isso significa que a distância em que a carga lançada explodiu do submarino foi de 4 Km.

O segundo lançamento já deve ser feito no sentido de verificar a posição do submarino numa dada direção. Considerando ainda a situação exemplificada anteriormente, suponha que queremos verificar profundidade. Basta manter os números N/S e L/O e alterar - para mais ou menos - o número P. Por exemplo, imagine que fizemos o lançamento [4] [1] [2] e que recebemos a dica 5.196. Isso significa que nos afastamos da posição procurada e é conveniente tentarmos outro sentido.

Se tentarmos o lançamento [4] [1] [4]

e obtivermos o mesmo número como dica (5.196), podemos então concluir que a profundidade do submarino é de 3 Km. Agora, devemos fazer o mesmo com os números N/S e L/O.

Depois de algumas partidas mal sucedidas, você achará o caminho certo para destruir o submarino. Eventualmente, após adquirir bastante prática, o leitor pode tentar alterar dois ou três números de uma só vez.

Vamos, agora, completar o programa de modo a sofisticá-lo um pouco. Antes, porém, analisemos suas partes principais:

Inicialização - Esta parte faz a apresentação na tela, recebe o nome de jogador através do teclado e sorteia a posição do submarino na região delimitada para o jogo. O sorteio é realizado através da função RND. Como a posição do submarino é dada através de três números, basta fazer um sorteio para cada um deles (veja as linhas 80 a 100). Se você quiser alterar a região onde o submarino pode estar, basta mudar os números 9 e 5 nas linhas 80, 90 e 100.

Lançamentos - Esta parte recebe os dez lançamentos do jogador. Cada lançamento necessita da introdução de três

```
10 COLOR 4,7:SCREEN 0:WIDTH 30
20 PRINT "BATALHA NAVAL TRI-DIMENSIONAL"
30 PRINT "DIGITE SEU NOME ALMIRANTE"
40 INPUT N$
50 CLS
60 LET LO=INT(RND*(TIME)*9)
70 LET NS=INT(RND*(TIME)*9)
80 LET P=INT(RND*(TIME)*5)
90 LOCATE 2,1:PRINT "ALMIRANTE",N$,N$,NS
100 LOCATE 2,4:PRINT "TIRO",L/O,N/S
110 P=INT(RND*(TIME)*5)
120 LOCATE 2,7:PRINT "DISTANCIA",P
130 FOR T=1 TO 10
140 LOCATE 3,T:PRINT T
150 LOCATE 11,T:PRINT
160 LINE INPUT LO$
170 LOCATE 17,T:PRINT
180 LINE INPUT NS$
190 LOCATE 22,T:PRINT
200 LINE INPUT P$
210 LET DLO=VAL(LO$)-LO
220 LET DNS=VAL(NS$)-NS
230 LET DP=VAL(P$)-P
240 LET D=INT(.5+1000*SGR(DLO*DLO+DNS*DNS+DP*DP)/1000)
250 LOCATE 26,T:PRINT
260 IF VAL(P$)4 OR VAL(P$)0 OR VAL(NS$)8 OR VAL(NS$)0 OR VAL(LO$)8 OR VAL(LO$)0 THEN PRINT "perdido!" GOTO 300
280 PRINT 0
290 IF D=0 THEN GOTO 300
300 NEXT T
310 FOR D=1 TO 10
320 FOR F=2 TO 15:COLOR 1,F:CLS:NEXT F
330 NEXT D
340 LOCATE 2,20
350 PRINT:PRINT:PRINT "SEU NAVIO ACABA DE SER ATINGIDO POR UM TORPEDO."
370 GOTO 450
380 FOR G=1 TO 10
390 FOR F=2 TO 15:COLOR 1,F:CLS:NEXT F
400 NEXT G
410 LOCATE 2,20
420 PRINT "PARABENS VOCE FOI PROMOVIDO A"
430 PRINT "... ALMIRANTE DE ESQUADRA."
440 PRINT
450 INPUT "VOCE FEZ '11-11' PONTOS"
460 RUN
```

Batalha Naval

SOM DE FUNDO,	
30 INTERVAL ON	
40 ON INTERVAL=150 GOSUB 470	
470 SOUND 6,8	
480 SOUND 7,55	
490 SOUND 10,9	
500 SOUND 11,0	
510 SOUND 12,180	
520 SOUND 13,10	
530 RETURN	
SOM DO LANÇAMENTO E DA EXPLOSAO DA CARGA,	
270 GOSUB 540	
540 SOUND 6,20	
550 SOUND 7,55	
560 SOUND 8,16	
570 SOUND 11,0	
580 SOUND 12,180	
590 SOUND 13,10	
600 RETURN	
SOM DE EXPLOSAO DO NAVIO,	
330 GOSUB 540	

Figura 3

números. Após receber cada um deles através do teclado, o programa calcula a distância entre a posição da explosão e a posição do submarino (através do Teorema de "Tritágoras"). Você lembra? Finalmente, os dados introduzidos e a distância calculada são impressos na tela no formato de uma tabela.

Finalização - Esta parte simplesmente encerra o jogo, informando a vitória ou derrota do jogador.

Agora sim, vamos incrementar o programa! Uma das grandes vantagens dos micros da linha MSX sobre os outros, está na facilidade com que podemos produzir sons através de seu BASIC. Acrescente ao programa as linhas que constam na figura 3. Todas elas tornam o programa um pouco mais atrativo. Agora, você pode começar a fazer suas próprias modificações acrescentando outros sons ou fazendo com que o navio e o submarino sejam desenhados na tela.

Uma mudança mais estrutural seria fazer com que, após cada tiro, o submarino se deslocasse. Isso tornaria o jogo muito mais difícil.

O programa deve ser digitado, mantendo-se a numeração das linhas da listagem. Posteriormente, você deverá completá-lo com as linhas que estão faltando.

Pierluigi Piazzi é formado em Química e Física pela USP. Foi professor do curso Angelo Vestibulares e já editou a revista Micro-hobby. É autor de vários livros e atualmente dirige a Aleph Publicações, responsável por muitos sucessos na área editorial, dada a sua orientação didática.

DADOS
sistemas de banco de dados

CURSOS DISPONÍVEIS

- Introdução à Microcomputação
- DOS - PC "Sistema Operacional"
- UNIX "Sistema Operacional"
- LINGUAGEM C "Ling. Programação"
- dBASE II "Programação Básica"
- dBASE II "Program. Avançada"
- dBASE III "Program. Básica"
- LOTUS 1-2-3 "Plan. Eletrônica"
- Framework "Sistema Integrado"
- Symphony "Sistema Integrado"
- Wordstar "Processador de Texto"

* Em desenvolvimento

MATERIAIS DIDÁTICOS: Publicações Técnicas desenvolvidas em português.

RECURSOS DIDÁTICOS: Conceitos e exemplos práticos, através de Micros e Telão de 72"

CURSOS FECHADOS E ABERTOS
NOVO ENDEREÇO
Rua Graeflândia, 1750 - Jardim Europa
CEP - 01434 - São Paulo - SP
Telefones: 881-4599 - 881-4032

"MIKROS"

- Microcomputadores Pessoais e Profissionais, Software, Suprimentos e Cursos.
- Financiamento em até 18 meses sem entrada e os preços mais baixos do mercado.
- Atendimento perfeito, profissionais treinados e habilitados para dar a você a certeza de um bom Investimento.

EQUIPAMENTOS

Micros das linhas: TRS-80 • Apple • IBM • Sinclair • TRS-80 Color Impressoras • Vídeos • Interfaces • Etc.

SUPRIMENTOS

Formulário Contínuo • Disquetes • Fitas • Mesas • Etiquetas • Etc.

SOFTWARE

Nacionais e Importados mais de 2.000 programas e jogos de todas as linhas.

CURSOS

Basic • Basic Avançado e DOS.

Av. Ataíde de Paiva, 566
sobreloja 211 e 202
Rio de Janeiro - R.J.
Tels.: (021) 239-2798 e 511-0599

Olhadinha

Francisco Corrêa

Trata-se de um jogo, do tipo educativo, para todos os computadores compatíveis com o TRS-Color. Com ele, crianças numa faixa de cinco a nove anos de idade poderão memorizar, brincando, a posição das teclas, bem como incrementar sua capacidade de concentração e acuidade visual.

O funcionamento do *Olhadinha* é muito simples: após o comando <RUN>, um menu aparecerá na tela, junto com uma simulação de um semáforo, à direita do vídeo. Um *bip* indicará cada mudança de luz, do amarelo ao verde. Logo em segui-

da, o programa fará aparecer, por um breve instante, uma palavra dentro de uma espécie de janela desenhada na tela. O desafio é digitar, no espaço indicado, a mesma palavra e conseguir abiscotar dez pontos pelo acerto.

Francisco J. M. Corrêa de Silva é Engenheiro Eletrônico e trabalha atualmente na Xerox do Brasil, como Especialista de Marketing. Lida com computação desde 1981 e é fundador do TRS-80 Color Club.

```

5 REM *****
6 REM * TRS-80 COLOR CLUBE RJ *
7 REM * C.POSTAL 2951 - 20001 *
8 REM * POR: FRANCISCO CORREA *
9 REM *****
10 CLS
20 PRINT@96,"A ESQUERDA DA TELA EXISTE UMA":PRINT
SINALEIRA IGUAL A DE TRANSITO."
30 PRINT"UMA PALAVRA SURGIRA POR UM INS- ANTE NA T
ELA E VOCE TERA DE DI- GITAR A RESPOSTA CORRETA."
40 PRINT"VOCE GANHA 10 PONTOS PARA CADA RESPOSTA
CORRETA. SE A RESPOSTA ESTIVER ERRADA, VOCE TERA U
MA SEGUNDA CHANCE, MAS NAO GANHARA PONTOS."
80 REM
90 PRINT@42," olhadinha ";
100 FORX=1 TO 200:NEXTX
110 SC=RND(0)
120 PRINT@42," OLHADINHA ";
125 SOUND200,I
130 FOR X=1 TO 200:NEXTX
135 PRINT@457,"QUALQUER TECLA";
140 IF INKEY#="" THEN 90
150 F=250
160 CLS 3
170 GOSUB 770
180 FOR X=1 TO 30
190 PRINT@X,CHR$(179);
200 PRINT@X+256,CHR$(188);
210 NEXT
220 FOR Y=32 TO 256 STEP 32
230 PRINT@Y,CHR$(181);
240 PRINT@Y+31,CHR$(186);
250 NEXT
260 PRINT@73,"*OLHADINHA*";
270 FOR X=136 TO 149
280 PRINT@X,CHR$(191);
290 PRINT@X+64,CHR$(191);
300 NEXT
310 PRINT@168,CHR$(191);:PRINT@181,CHR$(191);
320 T=T+1
330 IF T>10 THEN GOSUB 870
340 Z=RND(30)
350 IF Z=N THEN 340
360 N=Z
370 FOR F=1 TO N
380 READ N$
390 NEXT
400 PRINT@289," >>>DIGITE A RESPOSTA AQUI<<<"
410 REM
420 L$=CHR$(128)
430 FOR O=89 TO 219 STEP 32
440 PRINT@O,L$+L$+L$;:NEXTO
450 PRINT@418,"10 PONTOS PARA CADA ACERTO";
460 PRINT@122,CHR$(191);
465 SOUND10,I
470 GOSUB 940
480 PRINT@122,L$;
490 PRINT@154,CHR$(255);
495 SOUND10,I
500 GOSUB 940
505 SOUND10,I
510 PRINT@154,L$;
520 PRINT@186,CHR$(143);
530 GOSUB 940
540 PRINT@186,L$;
550 PRINT@169,N$;
560 FOR TL=1 TO F:NEXT
570 FOR B=169 TO 180
580 PRINT@B,CHR$(175);
590 NEXT B
600 GOSUB 780
610 PRINT@289," ":INPUT"QUAL A PALAVRA ";W$
620 IF W$<>N$ THEN GOTO700
630 IF TRY=1 OR TRY=2 AND W$=N$ THEN GOTO 670
640 IF W$=N$ THEN GOSUB 760
650 PRINT@300,"correto";
660 GOSUB 940
670 TRY=0 :GOSUB 770
680 RESTORE
690 F=F-50;GOTO320
700 PRINT@291,"a"L$"resposta"L$"esta"L$"errada..."
710 F=F+100
720 IF F>250 THEN F=250
730 TRY=TRY+1
740 IF TRY=2 THEN GOTO 800
750 GOTO 420
760 CO=CO+10
770 PRINT@481,"VOCE ATE AGORA TEM:"CO "PONTOS";
780 PRINT@288,""
790 RETURN
800 PRINT@302,"*":M=LEN(N$)
810 PRINT@290," A RESPOSTA E...";
820 FOR A=1 TO M
830 PRINT@306+A,M10*(N$,A,I);
840 SOUND RND(230),2:NEXT
850 GOSUB 940
860 TRY=0: RESTORE:GOTO 320
870 PRINT@355,"JOGA DE NOVO <S> OU <N>";
880 G$=INKEY$;IF G$=""THEN 870
890 IF G$="S" THEN 910
900 IF G$="N"THEN CLS4:END
910 RESTORE:F=250:CO=0:T=0
920 CLS:GOTO150
930 GOTO 320
940 FOR TY= 1 TO 600:NEXT TY
950 RETURN
960 DATA RISCAR,ACREOITAR,RUA,MACACO,DINHEIRO,PAO
970 DATA REAL,CADEIA,FLOR,PEDRA,SELO,MEL
980 DATA HOTEL,VILA,AEROPORTO,DAQOS,BATALHA,DESTE
990 DATA PLUMA,MALA,BARRO,ANEL,LAMPADA,FORCA
1000 DATA CAO,GATO,FRIJO,OURO,CARA,CORDA

```

Olhadinha

Micro
FichasTRS-COLOR
CÓDIGOS E MENSAGEM DE ERROMicro
Sistemas

- | | |
|---|--|
| 7/0 Tentativa de divisão por zero. | IE Tentativa de ler um dado além do fim do arquivo. Use as instruções EOF e LOF para descobrir se você já atingiu o fim de um arquivo, neste caso feche-o com (CLOSE). |
| AO Tentativa de reabrir arquivo. Aparece também se o RESET for pressionado durante operação com gravador. | IO Erro de entrada ou saída. Ocorre frequentemente nas operações de gravação com cassete. |
| BS Os índices de uma matriz estão fora da faixa. Use DIM para dimensionar a matriz.
Ex.: PRINT A(12), dará este erro se A não tiver sido pré-dimensionado com pelo menos 12 elementos. | LS String muito longa. Cada string pode conter no máximo 255 caracteres. |
| CN Tentativa de usar o comando CONT em momento inadequado.
Ex.: Após o processamento da última linha do programa. | NF NEXT sem FOR. Foi utilizada uma instrução NEXT sem a respectiva instrução FOR. |
| OO Tentativa de redimensionar uma matriz. Cada matriz só pode ser dimensionada uma vez.
Ex.: Não seria possível usar DIM A(10) e DIM A(50) no mesmo programa. | NO Tentativa de enviar dados para um arquivo ainda não aberto (OPEN). |
| ON Uso ilegal de número de dispositivo. Com OPEN; CLOSE; PRINT ou INPUT você só pode usar 0, -1 ou -2. | OO READ sem OATA. Os dados das instruções OATA são insuficientes para a instrução REAO. |
| OS Instrução direta no arquivo de dados sem a especificação do número de linha. Pode ser causado se você tentar carregar um arquivo de dados em fita. | OM Falta memória. Toda a memória disponível foi usada ou reservada. |
| FC Chamada incorreta de função. Foi usada uma função com um valor ilegal nos seus parâmetros.
Ex.: SOUND (260, 260) ou A(-1). | OS Faltou espaço para strings. Ove-se limpar espaços com CLEAR. |
| FD Formato de campo incorreto. Ocorre quando se lê (INPUT) ou grava (PRINT) um dado usando formato inadequado.
Ex.: INPUT #1,A quando o dado no arquivo é uma string. | OV Estouro de variável numérica. Foi gerado um número muito grande ou muito pequeno que o computador não tem capacidade para manipular. |
| FM Modo de arquivo inválido. Ocorre quando se tenta entrar com dados em um arquivo aberto apenas para leitura (L) ou imprimir dados de um arquivo aberto para entrada (I). | RG RETURN sem GOSUB. Foi encontrada uma instrução RETURN, sem um GOSUB correspondente. |
| ID Instrução direta ilegal.
Ex.: Tentar usar INPUT sem numeração de linha. | SN Erro de sintaxe. Uso incorreto de pontuação; parênteses; aspas; caractere ilegal ou comando mal digitado. Redigite a linha ou comando. |
| | ST Formula de string muito complexa. Foi usada uma operação muito complexa dentro de uma string. Oviada em operações menores para funcionar. |
| | TM Incompatibilidade de tipo de variável. Foi tentada uma atribuição incompatível com a string utilizada. Ex.: A = "dado". |
| | UL Linha indefinida. Foi solicitado um desvio (GOTO; GOSUB), para uma linha inexistente. |

ATI Produção

Economize tempo e dinheiro.

Racionalize seu 'business forms'

A EDITORA JOLAN acaba de preencher uma lacuna do mercado de publicações especializadas, com o lançamento do livro **MANUAL TÉCNICO DE FORMULÁRIOS CONTÍNUOS**, de Luiz Carlos Cardoso, um dos mais renomados profissionais do ramo.

Um verdadeiro curso de especialização em "business forms", o livro **MANUAL TÉCNICO DE FORMULÁRIOS CONTÍNUOS** é indispensável para usuários de informática, profissionais de administração, O&M, processamento de dados, analistas de sistemas, programadores, vendedores e compradores de Formulários Contínuos e técnicos de empresas. Ensina como projetar o formulário em função do seu equipamento de processamento, tendo em vista também a racionalização dos custos industriais do próprio formulário.

MANUAL TÉCNICO DE FORMULÁRIOS CONTÍNUOS contém 248 páginas impressas em papel off-set, com 206 ilustrações, fotos e tabelas, no formato 18x27cm, capas a 4 cores em papel especial, com acabamento costurado e encadernado.

Por se tratar de um livro de caráter técnico-didático, a aquisição de exemplares por empresas ou pessoas físicas possibilita desconto no Imposto de Renda, além de incentivos fiscais por apoio cultural e técnico.

Manual Técnico de
Formulários ContínuosPreço por exemplar
Cr\$ 250.000

Validade: até 28/2/86

Desejo receber () exemplar(es) do livro Manual Técnico de Formulários Contínuos, pelo(s) qual(is) pagarei a quantia de Cr\$ _____ (_____).

Nome _____
Endereço _____
Cidade _____ Estado _____ CEP _____

Para tal estou enviando um cheque nominal à Editora JOLAN, com despesas de entrega incluídas.

Editora JOLAN Ltda.

Av. Rio Branco, 156 - sala 1537 - Tel.: 262-4619 - CEP 20040
Rio de Janeiro, RJ.



Edgar Athayde Meneghetti

Este jogo foi escrito para os micros da linha Sinclair que possuem 16 Kb de memória, pois ele ocupa cerca de 3 Kb e é totalmente elaborado em linguagem de máquina. Nele, você comanda o submarino Polaris em águas inimigas, onde o objetivo é fazer o maior número de pontos sem ser destruído.

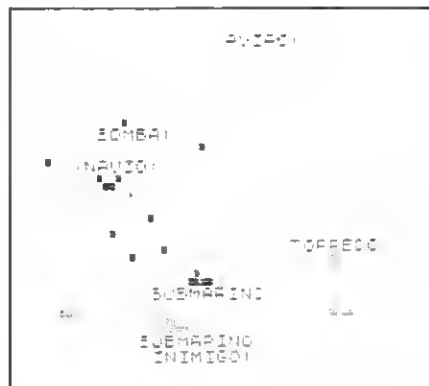
Inicialmente, carregue o MICROBUG e crie uma linha REM com 2542 caracteres. Introduza os códigos hexadecimais a partir do endereço 16509. Após ter digitado tudo, basta dar RUN que o programa será gravado, aparecendo em seguida a abertura. Para iniciar o jogo, pressione NEWLINE ou o botão de tiro do joystick que a batalha irá começar!

A tabela de pontos é mostrada na figura 1.

Avião = 10 pontos
Barco = 50 pontos
Submarino inimigo = 100 pontos
Líder da tropa = 500 pontos
Mina = não pode ser destruída
Bomba = não vale pontos

Figura 1

```
16509 00 01 4E 01 EA 0E 4C 98
16517 80 CC 40 4A 4D 29 4D 08
16525 4D 7F 4A 0C 4A C1 FE 80
16533 08 FE 08 08 08 08 08 08
16541 00 C2 84 44 C5 C9 2A 82
16549 40 36 81 23 36 83 CD 08
16557 02 23 08 28 08 4C C3 08
16565 49 C8 5C 28 11 C8 64 28
16573 24 C8 6C 08 C8 65 28 43
16581 C0 50 28 05 C9 06 2A 02
16589 40 23 23 7E CD 92 40 28
16597 28 36 80 23 22 82 40 36
16605 01 23 36 83 C9 2A 82 40
16613 11 DF FF 19 7E CD 92 40
16621 23 7E CD 92 40 28 22 82
16629 40 11 21 00 19 36 00 23
16637 36 00 2A 82 40 36 81 23
16645 36 83 C9 2A 82 40 11 21
16653 00 19 7E CD 92 40 23 7E
16661 CD 92 40 28 22 82 40 11
16669 DF FF 19 36 00 23 36 00
16677 2A 82 40 36 81 23 36 83
16685 C9 2A 82 40 28 7E CD 92
16693 40 22 82 40 23 36 00 23
16701 36 00 28 36 83 28 36 01
16709 C9 00 2A 82 40 11 DF FF
16717 19 7E FE 00 20 0C FE 08
16725 28 F6 FE 00 20 20 36 18
16733 18 EE 0E 04 06 FF 10 FE
16741 00 20 F9 11 21 00 19 7E
16749 FE 08 28 FA FE 81 28 04
```



Pode-se jogar pelo teclado ou pelo joystick. E quando aparecer a abertura, o programa poderá ser brechado pressionando SHIFT e BREAK simultaneamente.

Na figura 2 encontramos um resumo dos principais endereços para eventuais modificações ou até mesmo uma personalização do jogo. Nela, observamos ainda que o número de bombas é a proporção de 0 a 254.

Se você quiser modificar o recorde, altere os códigos de 18021 a 18025;

- 18985 — Número de bombas lançadas pelos aviões (quanto maior o número mais bombas).
- 18990 — Número de submarinos (quanto maior o número mais submarinos).
- 18995 — Número de minas soltas pelos submarinos (quanto menor o número mais minas).
- 19000 — Número de bombas soltas pelo líder (quanto maior o número mais bombas).
- 17441 — Valor atribuído aos aviões.
- 17450 — Valor atribuído aos submarinos.
- 17469 — Valor atribuído aos barcos.
- 16736 — Valor da pausa entre a subida do tiro e o seu apagamento (caso o tiro esteja pouco visível, aumente o valor para 6 ou mais).

Figura 2

e para mudar o número de vidas, modifique o valor do endereço 17898 da seguinte forma: ao número de vidas desejadas, some 28.

Edgar Athayde Meneghetti cursa a segunda série do Colégio de Aplicação, de Porto Alegre. É um autodidata que aprendeu a programar em BASIC e Assembler através da leitura de livros e revistas.

Polaris

```
17501 40 7E FE 25 28 05 34 10
17509 F5 E1 C9 36 1C 28 7E FE
17517 25 28 F8 34 10 E8 E1 C9
17525 2A 82 40 7E FE 81 20 07
17533 23 7E FE 83 20 01 C9 2A
17541 90 40 7E FE 1C 28 04 35
17549 C3 FF 44 C3 81 45 00 C9
17557 00 00 00 00 00 00 00 00
17565 00 76 00 05 18 01 EA 2A
17573 0C 40 06 09 23 7E FE 76
17581 28 05 FE 12 C8 18 F5 10
17589 FC C3 00 4A 36 00 23 36
17597 00 CD 04 41 C9 CD 75 44
17605 CD A3 40 CD 7E 41 CD 08
17613 43 CD DC 43 CD 87 42 CD
17621 75 44 CD A3 40 CD 7E 41
17629 CD 08 43 CD 78 43 CD 21
17637 43 CD 75 44 CD 43 40 CD
17645 7E 41 CD 08 43 CD 87 42
17653 CD A4 44 CD 75 44 CD A3
17661 40 C9 06 20 2A 82 40 36
17669 81 23 36 83 C5 2A 82 40
17677 01 DF FF 11 21 00 23 7E
17685 C6 80 77 28 7E C6 80 77
17693 20 7E C6 80 77 19 7E C6
17701 80 77 23 7E C6 80 77 23
17709 7E C6 80 77 23 7E C6 80
17717 77 09 7E C6 80 77 09 7E
17725 C6 80 77 28 7E C6 80 77
17733 28 7E C6 80 77 28 7E C6
17741 80 77 06 0A 0E FF 0D 20
17749 FD 10 F9 C1 10 02 18 03
17757 C3 09 45 2A 0C 00 01 94
17765 02 09 06 03 23 7E FE 76
17773 28 0F FE 00 20 07 23 7E
17781 FE 09 28 03 28 36 00 18
17789 E8 10 FC C9 2A 8C 40 11
17797 76 01 19 36 80 23 36 AE
17805 23 36 82 23 36 80 23 36
17813 A9 23 36 AA 23 36 80 23
17821 36 AF 23 36 84 23 36 AC
17829 23 36 84 E1 E1 C3 10 46
17837 00 00 00 00 00 00 00 00
17845 00 00 00 00 00 76 00 06
17853 10 01 EA 18 0C E1 7E 23
17861 E5 FE 43 C8 CD 08 08 18
17869 F4 CD C2 45 35 34 33 39
17877 34 38 00 1C 1C 1C 1C 1C
17885 1C 00 37 2A 28 1C 1C 1C
17893 1C 1C 1C 1C 00 1F 00 81
17901 83 00 00 9C 43 2A 0C 40
```

Polaris

SOFTWARE NASAJON



Faz a Folha de Pagamento de sua empresa, emitindo relatórios como Guia de IAPAS, Guia de FGTS, Relação de Empregados, Relação para I.R., Relação para Banco, Informe de Rendimentos, Acumulados Anuais, RAIS e Recibo de Pagamento. A folha pode ser semanal ou mensal. As tabelas são modificadas pelo próprio usuário.

Permite também, adiantamentos de salário, reajuste salarial, alterações de acumulados e outras funções que agilizam o processamento da Folha de Pagamento da empresa.



A Contabilidade de um mês em apenas 2 horas! Este Sistema permite o cadastramento de históricos padronizados e de plano de contas com até 5 níveis. Emite Diário, Razão, Balancete, Balanço, Demonstração de Resultados, Demonstração de Lucros e Prejuízos acumulados, Listagem por centro de custo e extrato de contas, entre outras funções.

Compatíveis com as linhas TR5-80 e Apple.



Controle o estoque de itens com Especificação, Estoque Mínimo, Unidade, Fornecedor, Localização e outras informações relacionadas no item como Custo Médio, Entradas e Saídas no período, etc. Fornece Listagens Geral e Parcial dos produtos, Listagem Físico-Financeira, Listagem dos produtos abaixo do estoque mínimo, Liste de Preços e Etiquetas, entre outros. Admite também, Reajuste de Preços, Alteração de Dados e Exclusão de Produtos.

A NASAJON oferece assistência técnica total, garantia permanente e mantém à sua disposição programadores e analistas para desenvolver sistemas específicos sob encomenda. Conte com a NASAJON SISTEMAS.



Av. Rio Branco, 45 - Grupo 1.311
Rio de Janeiro - CEP 20.090
Tels.: (021) 263-1241 e 233-0615



Aldo Barduco Jr. e Pierluigi Piazzi

O peixe guloso

No fim do dia, quando os pescadores se retiram, deixam latas com minhocas e outras iscas na beira do lago. Water, um peixe de nome científico *PISCES FAMINTUS*, está com muita fome e sai à procura de alimento. Tendo o hábito de ficar soltando bolhas na água, descobre que, ao soltá-las sob as latinhas, destas caem minhocas.

Os peixes desta espécie habitam os frios lagos do Tibet, onde existe o terrível *LUPUS AQUATICUS*, popularmente conhecido como Peixe-lobo. Os Peixes-lobos têm como dieta tudo que encontram pela frente, menos as terríveis Piranhas Azuis.

O único peixe capaz de se alimentar

destas Piranhas é Water, mas, se a Piranha estiver por trás, o devorará em poucos instantes (a "ecologia" da região é realmente confusa!).

O Tibet tem um relevo jovem (terciário), por isso são tão comuns desabamentos e terremotos. A partir de uma certa hora, os desabamentos dos barrancos do lago começam e não param mais.

Como os gatos têm sete vidas, Water possui três.

Boa sorte e muito cuidado. No Tibet, os mistérios são muitos. . .

O JOGO

Você deverá conduzir Water, desviando-o dos Peixes-lobos (e da Piranha, se

estiver atrás). As teclas para controle de Water são: 7 (para subir); 6 (para descer) e 0 (para soltar bolhas). No topo da tela, estão indicados os pontos e as vidas.

As bolhas só podem ser soltas uma de cada vez. Caso a bolha atinja a latinha, minhocas cairão na água, e Water deverá pegá-las. Já, se ela atingir o louco Monge Nadador, esta sumirá.

Isacas amarelas de formas estranhas aparecerão na tela. Ao pegá-las, você adiciona 1000 pontos ao seu placar.

Os Peixes-lobos virão à sua frente em ziguezague. Desvie, pois, se não o fizer, a morte será certa. Já as Piranhas Azuis vêm por trás e tentam pegá-lo. Se você,

porém, pegá-la por trás, fará muitos pontos.

Ao passar dos 1000 pontos, uma pedra começará a cair da borda do lago e, se atingir Water, este perderá uma vida. Ao final, o programa mostrará os recordistas. Caso você tenha batido algum recorde, deverá introduzir seu nome, que será exposto à admiração pública e exaltação generalizada!

PARTES IMPORTANTES DO PROGRAMA

Inicialmente, explicaremos as rotinas mais interessantes e importantes do programa Water e, a seguir, daremos algumas dicas de alterações.

A linha 10 faz a leitura dos DATA, que estão a partir da linha 9100 e coloca seus valores na memória do UDG redefinível pelo usuário (dizemos "redefinível pelo usuário", pois existem o UDG 1 e o UDG 0 que o próprio TK define).

A linha 11 dimensiona R\$ para que guarde o recorde e os recordistas.

Na linha 30, o primeiro PRINT executa uma impressão na parte inferior da tela, nas linhas de edição.

Na linha 50, iniciamos o movimento do Water, pois fazemos um LOOP de 2 até 31 e armazenamos o valor em x (que corresponde à coordenada x da posição do peixe). Nesta mesma linha, colocamos em O o valor 1, se O for 0, e 0, se O for 1. Daqui a pouco você verá o porquê.

A linha 70 faz a leitura do teclado.

Na linha 71, verificamos se existe algum caráter de cor diferente do fundo, à frente de Water. Este recurso nos permite saber qual caráter está na dada po-

sição (ver da linha 1000 até a 1025).

Na linha 75, verificamos se Water mudou de linha (subiu ou desceu) e, em caso afirmativo, apagamos a velha posição que ele ocupava.

Na linha 80, imprimimos Water e variamos sua cauda de acordo com o valor de O (veja linha 50).

A linha 85 verifica se estamos quatro caracteres atrás da piranha e, se estivermos, faz com que Water abra e feche a boca. Já a linha 101 verifica se a pedra atingiu Water.

Da linha 105 até a linha 210 há o movimento de todos os outros elementos na tela e a verificação de se algum destes atingiu Water. Não entramos em detalhes, pois os processos já foram mencionados (verificação de uma posição no vídeo, verificação nas próprias variáveis que contêm a posição do Water).

Da linha 2000 até a 2040 há a exposição dos nomes recordistas, verificando se houve recorde batido. Em caso positivo, é feita a inserção do nome no novo recordista. As linhas 9000, 9010 e 9020 são continuação desta rotina (2000-2040), sendo apenas de recursos sonoros e visuais do final do programa.

Na linha 3000, usamos recursos de desenho (PLOT, DRAW), para escrever "W" e desenhar a lua.

Na linha 3060, guardamos, na variável O\$, a melodia que toca durante a apresentação.

Na linha 3070, é impressa a assinatura do autor do jogo e o ano em que foi feito (Aldo 85).

As possíveis mudanças que não alterariam a estrutura do programa são:

Modificar as teclas para 8 e 9, ao in-

vés de 6 e 7, seria muito interessante para os que possuem joystick.

Para modificar o som do programa, altere a linha 95.

Se achar que o número inicial de vidas é muito pouco, altere o FOR V=3 ... para FOR V=Número de vidas iniciais desejado...

Não mude a cor do jogo, pois isto requer um conhecimento maior sobre o funcionamento das funções ATTR que, no caso, serão totalmente alteradas (em termos de valores referidos aos caracteres do jogo).

Para se gravar o programa, basta dar o comando direto SAVE "WATER" LINE 10 e o mesmo irá entrar em auto-start. Se quiser gravá-lo de uma forma mais compacta, execute a linha 10 e depois elimine as linhas de 9100 até o final e substitua a linha 10 por LOAD "" CODE. Em seguida, grave o programa em duas etapas:

- 1) SAVE "WATER" LINE 10
- 2) SAVE "WATER" CODE USR "a", 168

Quanto ao resto, deixamos por conta de sua habilidade e imaginação. Bom divertimento!

Aldo Barduco Jr. está cursando a 8ª série do primeiro grau e é autodidata em BASIC a Assembler. Possui um TK 85 e um TK90X.

Pierluigi Piazzi é formado em Química e Física pela USP. Foi professor do curso Anglo Vestibulares e já editou a revista "Microhobby". É autor de vários livros e atualmente dirige a Aleph Publicações, responsável por muitos sucessos na área editorial, dada a sua orientação didática.

```

5)REN Aldo Barduco Jr
10 FOR F=USR "a" TO USR "u":7:
READ X: POKE F,X: NEXT F
11 LET A$="" LET B$="" FOR F
=1 TO 32 STEP 2: LET A$=A$+CHR$
140: LET B$=B$+CHR$ 149+CHR$ 140
3: NEXT F: LET A$=A$+A$: DIM R$(1
0,20): DIM R(10)
12 BORDER 1: PAPER 1: INK 7: G
OBUB 3000
20 BORDER 1: CLS: LET Y=19: L
ET BX=0: LET BY=0: LET B0=0: LET
0=0: LET NX=0: LET NY=3: DIM N(
3): LET P=0: LET T=0: LET TC=0:
LET V=0: LET Y=0:
30 PRINT #1: " BRIGHT 1:CH
R$ 144:CHR$ 145: BRIGHT 0: Water
r, o peixe guloso": LET MY=4: LE
T MX=0: PRINT AT 21,0:AS:AT 21,0
: INK 4: OVER 1:85
40 LET K=0: LET PX=0: LET PY=1
0: LET DX=0: LET DY=10: FOR F=0
TO 2: PRINT AT F,0: PAPER 5:
NEXT F
45 LET TY=Y: PAPER 5: INK 0: P
RINT AT 1,5:CHR$ 146:AT 1,15:CHR$
146:AT 1,25:CHR$ 146:AT 2,5:CHR$
147:AT 2,15:CHR$ 147:AT 2,25:
CHR$ 147: PAPER 1: INK 7
46 PRINT AT 0,3: PAPER 5: INK
0: "Pontos": PAPER 5: INK 0:AT 0
,10:P
49 FOR V=3 TO 1 STEP -1: PRINT
AT 0,16: PAPER 5: INK 0: "Vidas":
1: PAPER 5: INK 0:V: SOUND 10,2
0: PRINT AT 3,0:
50 FOR X=2 TO 31: LET O=0+(0=0
)-(0=1)
70 LET Y=Y+(INKEY$="6" AND Y<1
9)-(INKEY$="7" AND Y<5): IF NOT
80 THEN LET B0=(INKEY$="0"): LET
BY=(Y-1 AND 80): LET BX=(X-2 AND
80): IF 80 THEN PRINT AT BY,BX
:CHR$ 151
71 LET K=X+(ATTR (Y,X+1)<15)
75 IF TY<Y THEN PRINT AT TY,X
-3: " LET TY=Y
80 PRINT AT Y,X-2: " +CHR$ 14
4+CHR$ 145 AND 0=1)+(CHR$ 150+CH
R$ 145 AND 0=0)
85 IF PX=4X AND X(PX THEN PRI
NT AT Y,X-2: " :CHR$ 144:CHR$ 15
90 IF K THEN LET K=0: GOTO 100
95 SOUND 100,12
100 IF P<1000 THEN PRINT AT NY,
NX: " : LET NY=NY+(NY<20)-(NY-4
AND NY<19): PRINT AT NY,NX:CHR$
150: SOUND 100,15
101 IF (NX=X AND NY=Y) OR (NX=X
-1 AND NY=Y) THEN PRINT AT Y,X-1
: " : NEXT V: GOTO 2000
105 IF NY=4 THEN PRINT AT NY,NX
: " : LET NX=INT (RND*31)
110 IF N(1) AND (PX=X-1 AND PY=
Y) THEN PRINT AT Y,X-2: " : NE
XT V: GOTO 2000
120 IF T THEN LET T=T-1
130 IF BY=NY AND BX=NX THEN LET
B0=0
140 LET NX=NX-(NX<0)+(31 AND NX
=0): PRINT AT NY,NX:CHR$ 157 AND
DX/2=INT (NX/2)+(CHR$ 156 AND
NX/2=INT (NX/2): "
150 IF 80 THEN PRINT AT BY,BX: "
: LET BY=BY-(BY<3): PRINT AT B
Y,BX:CHR$ 150
160 IF N(1) THEN PRINT AT PY,PX
: " : LET PX=PX+(INT (RND=2) AND
PX<31)-(30 AND PX<30): LET PY=P
Y+80N (Y-PY)+INT (RND=2): PRINT
AT PY,PX: INK 5:CHR$ 153
170 IF N(2) THEN PRINT AT DY,DX
: " : LET DX=DX-1: LET DY=DY+80N
(Y-DY-RND*1,5): PRINT AT DY,DX:
INK 4:CHR$ 155: IF NOT DX THEN
PRINT AT 0Y,DX: " : LET N(2)=0
180 IF NOT N(1) AND RND<1 THEN
LET PX=0: LET N(1)=PI/PI
195 IF NOT N(3) AND RND<0,02 THEN
N LET N(3)=1: PRINT AT RND*15+5,
RND*29+1: INK 4:CHR$ 154
200 IF NOT N(2) AND RND<2 THEN
LET DX=31: LET N(2)=PI/PI
205 IF (DX=X-1 AND DY=Y) OR (DX
=X AND DY=Y) THEN PRINT AT Y,X-2
: " : LET OY=(21-4): NEXT V: D
OTO 2000
206 IF BY=3 THEN LET BY=2: LET
B0=0: PRINT AT 3,BX: " : DOTO 0+
(900 AND ATTR (BY,BX)=40)+(207 A
ND ATTR (BY,BX)=40)
207 IF BY=3 THEN PRINT AT BY,BX
: " : LET B0=0
208 IF NOT NX THEN PRINT AT NY,
0: "
210 NEXT X: PRINT AT Y,30: " :
AT TY,30: " : GOTO 50
900 LET L=INT (RND*5+1): FOR F=
1 TO L: LET L7=INT (RND*7+3): PR
INT AT L7,F+BX:CHR$ 152: NEXT F:
LET T=30: SOUND 100,10
910 GOTO 207
1000 IF ATTR (Y,X+1)=11 THEN LET
P=P+Y: PRINT AT Y,X-1:CHR$ 14
4:CHR$ 159: PAPER 5: INK 0:AT 0,
10:P
1010 IF ATTR (Y,X+1)=13 THEN LET
N(1)=0: LET P=P+60+INT (RND*20)
: PRINT PAPER 5: INK 0:AT 0,10:P
1020 IF ATTR (Y,X+1)=12 THEN NEX
T V: GOTO 2000
1025 IF ATTR (Y,X+1)=14 THEN LET
P=P+1000: PRINT PAPER 5: INK 0:
AT 0,10:P: PRINT AT Y,X+1: " : B
OUND 0,15
1030 DOTO 100
2000 PRINT AT 2,1: "OS RECORDISTA
B: " : FOR F=4 TO 13: PRINT AT F,2
:RS(F-3):R(F-3): NEXT F
2010 FOR F=1 TO 10
2020 IF R(F)<P THEN LET R1=F: GO
TO 2040
2030 NEXT F: GOTO 9000
2040 FOR F=9 TO 1 STEP -1: LET
R$(F-1)=R$(F): LET R(F)=R(F):
NEXT F: PRINT AT 21,2: "Recordist
a digite seu nome: " : INPUT R$(1
) : LET R(1)=P: FOR F=4 TO 13: P
RINT AT F,2:RS(F-3):R(F-3): NEXT
F: GOTO 9000
3000 CLS: PLOT 244,171: DRAW -1
,-10, 2: DRAW 2,10,3: PRINT AT 5
,0: INK 5:85
3010 PRINT: PRINT INK 3: " :CN
R$ 144:CHR$ 145
3020 PLOT 93,170: DRAW 3,-20: DR
AW 2,15: DRAW 2,-15: DRAW 3,20:
PRINT AT 2,13: " A T E R "
3030 PRINT AT 16,0: " :CHR$
164: " By Aleph Publicações
3035 PRINT AT 18,0: " (P) By
Aldo Barduco Jr
3040 PRINT AT 21,0: "Aperte uma l
etra para continuar"
3045 IF INKEY$="" THEN GOTO 3045
3047 PRINT AT 21,0: "
3050 PRINT AT 9,0: " Controle Wat

```

Freeway data buffer



APROVADO PELA SEI
Portaria 205 de 23-08-85

Aumente a produtividade do seu micro, acabando com a espera pela impressora.

- FREEWAY é um "data buffer" que controla sozinho a impressora durante a emissão dos seus relatórios, liberando o micro para outras tarefas.
- Memória de 64 Kb expansível a 576 Kb.
- Opera em modo comprimido, permitindo armazenar relatórios muito maiores que sua memória.
- 9 funções: COPIAS ADICIONAIS SEM USO DO MICRO, AVANÇO DE PÁGINA, PAUSA AUTOMÁTICA POR PÁGINA, ESTATÍSTICAS, MODO DIRETO, AUTOTESTE, etc.
- Comunicação serial (RS 232), paralela (Centronics) e combinações, permitindo ligar micro serial em impressora paralela e vice-versa.
- Pode ser ligado em terminais, plotters, RJE, etc.

Rally

Luís Mauricio Chopard Bonilauri

Rally/Bas é um jogo rápido e emocionante, que poderá diverti-lo por longas horas. Ele foi programado em BASIC, mas possui duas rotinas em linguagem de máquina, para agilizá-lo e produzir som.

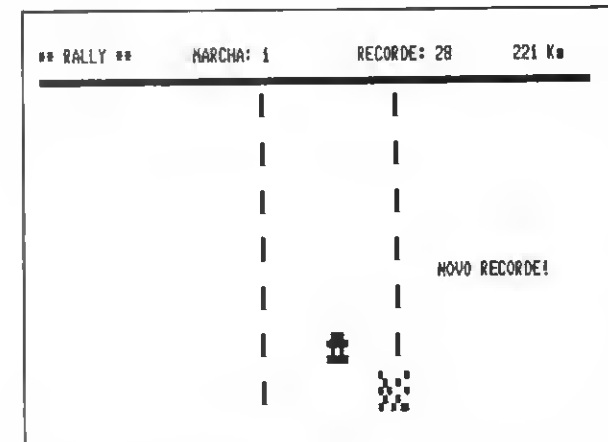
Originalmente, o programa foi feito em um CP 500, com drive, e para rodá-lo em equipamentos com cassete é necessário uma modificação, já que ele possui duas rotinas em Assembler, e o monitor BASIC residente só suporta uma. A modificação consiste em suprimir a rotina de som, uma vez que a outra rotina serve para deslocar a tela, sendo por isso mais importante. Deve-se proceder da seguinte maneira:

1º) Alterar a linha 80 para: **80 POKE 16526,0:POKE 16527,125.**

2º) Retirar as linhas 120, 130 e 180.

3º) Retirar da linha 340 o trecho: **FOR Y=1 TO 5:X=USR1(50):NEXT.**

4º) Mudar a linha 470 para **470 REM.**



Exemplo de tela do jogo.

Luís M. C. Bonilauri tem 13 anos e, desde 1984, faz programas em BASIC e Assembler. Aprendeu as duas linguagens através de livros e manuais.

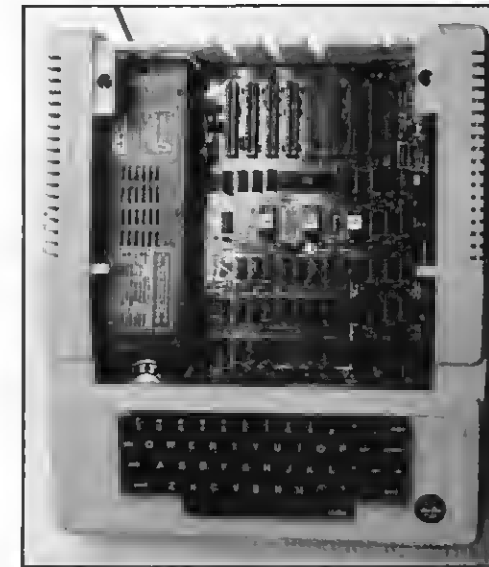
```

10 *****
20 * LUIS MAURICIO CHOPARD BONILAURO, RIO DE JANEIRO R.J. *
30 *****
40 CLEAR 300
50 RE=0
60 RANDOM
70 A%=CHR$(174)+CHR$(175)+CHR$(132)+CHR$(26)+STRING$(3,H)+CHR$(
31)+CHR$(131)+CHR$(129)
80 DEFUSR=32000:DEFUSR1=-8192
90 B%=CHR$(170):C%=CHR$(149)
100 F=32000
110 READ A:POKE F,A:F=F+1:IF A<201 THEN GOTO 110
120 F=-8192
130 READ A:POKE F,A:F=F+1:IF A<201 THEN GOTO 130
140 GOSUB 400
150 GOSUB 500:CLS:PRINT@0,"** RALLY **" MAPLHA: 1
RECORDE=:RE:264,STRING$(64,140):
160 P=925:M=1:T=0
170 DATA32,127,63,126,17,255,63,18,1,0,3,237,184,33,154,60,54,
,17,155,60,1,14,0,237,176,33,217,60,54,32,17,218,60,1,14,0,237,
76,201
180 DATA205,127,10,77,68,62,1,105,211,255,45,32,253,60,105,111,
55,45,32,253,13,16,238,175,211,255,201
190 GOSUB 370
200 PRINT@25+128,B%:241+128,C%:
210 A=USR1(0)
220 IF PEEK(15350)=64P=P+2 ELSE IF PEEK(15350)=16P=P-2
230 IF PEEK(15360+P)>128 OR PEEK(15362+P)>128 GOTO 310
240 PRINT@P,A%:
250 IF PEEK(14352)=2 THEN NM=1:PRINT@26,"1":ELSE IF PEEK(14352)
4 THEN NM=0:PRINT@26,"2":
260 IF RND(0)=1 PRINT@P-75B,A%:
270 I=1+1
280 IF M=0 THEN GOTO 200
290 FOR F=1 TO 25:NEXT
300 GOTO 200
310 PRINT@54,I:"Km":
320 PRINT@92,STRING$(15,128):2486,STRING$(15,128):
330 PRINT@P-64,CHR$(144):" ":CHR$(144):20P,CHR$(166):21HP,CHR$(
HR$(145):CHR$(132):20P+64,CHR$(129):CHR$(130):CHR$(130):LH$(145)
:
340 IF I>RE THEN RE=I:PRINT@760-128-10,"NOVO RECORDE: ":RE:1-1
TO 5:X=USR1(50):NEXT ELSE FOR C=1 TO 300:NEXT
350 PRINT@89B,"TECLE (ENTER) P/":2065,"RECORDE:":
360 IF PEEK(15350)=0 THEN 360 ELSE GOTO 150
370 FOR F=2 TO 15 STEP 2:PRINT@F*64+25,CHR$(170):20P+64+41,CHR$(1
49):NEXT:PRINT@P,A%:
380 FOR F=0 TO 20:PRINT@54,"ATENCAO ":FOR I=1 TO 35:PRINT@5
4,STRING$(7,128):FOR I=1 TO 34:NEXT:NEXT
390 RETURN
400 CLS:PRINT@23,"* * R A L L Y * *":20H7,STRING$(19,"=")
410 PRINT:PRINT:PRINT" Dirija seu carro em uma pista de Rally,
mas cuidado com os outros concorrentes!"
420 PRINT" Use ( e ) para movimentar seu carro, e ( e ) para m
gatar a primeira ou segunda marcha, respectivamente. SPRINT
430 PRINT" BOA SORTE!!!"
440 PRINT:PRINT:PRINT" Por: Luis Mauricio Chopard Bonilauri
450 PRINT@977,"TECLE (ENTER) PARA COMEÇAR A CORRIDA:
460 F=30
470 X=USR1(30030+F):F=F+10:IF F=150 THEN 140
480 IF PEEK(15350)=0 THEN 470
490 RETURN
500 CLS:PRINT@519,"Qual o nível (dificuldade) que deseja jogar?
510 IF V>21 OR V<5 THEN 500
520 RETURN

```

Rally/Bas

O UNITRON REVELA O SEU SEGREDO:



MODULARIDADE

Quem compra um micro está investindo no presente e no futuro. Por isso o equipamento deve acompanhar a evolução das necessidades e aplicações do usuário. Através dos seus oito slots (parte superior da foto), o UNITRON APII permite a expansão do equipamento na medida em que crescem as necessidades e de acordo com as economias de cada um.

Você pode começar comprando um UNITRON APII e ligá-lo na TV (P&B ou cores). Depois pode acrescentar disk-drives, impressora, monitor profissional, placa Z-80 para CP/M, placa 80 colunas, interface de co-

municações, etc., etc. Você pode também aumentar a memória do seu micro com placas de 16, 32, 64 ou 128K. Tudo isto de modo bastante simples, com manuais em português e fácil instalação. Quanto ao software, você dispõe de programas como o VISICALC, DBASE, WORD STAR e milhares de outros best-sellers, desenvolvidos para o "APPLE II+" norte-americano. Essa modularidade e riqueza de software é que fazem do UNITRON APII o micro mais vendido para empresas, profissionais liberais e famílias inteligentes. Fique com o melhor. Fique com o UNITRON APII.

O MELHOR CONJUNTO

UNITRON APII, MONITOR COMPO, DRIVE ELEBRA, IMPRESSORA MONICA E INTERFACES.

2690RTN's válido até 30/01/86

O MAIOR REVENDEDOR UNITRON DE TODO O PAÍS

RUA SETE DE SETEMBRO, 99 - 11º ANDAR
TEL.: PABX (021) 224-7007 - CEP 20050 - RIO DE JANEIRO - RJ
AV. PAULISTA, 726 - 5º ANDAR - CONJ. 506
TEL.: (011) 288-4492 - CEP 01362 - SÃO PAULO - SP

compu
micro
INFORMÁTICA EMPRESARIAL LTDA.

William Tadao Tanji

Agora, é só apertar **RUN...** e boa sorte.

William Tadeo Tanji é formado pela Escola Técnica Senai Sulco-Brasileira, de São Paulo, em Mecânica de Precisão. Atualmente estuda no Instituto de Ensino de Engenharia Paulista. Tem a microcomputação como hobby, sendo usuário de um CP-200.

```
105 LET A$=INKEY$
106 LET I=f+.5
107 PLOT I=2+11.32
108 IF I=21 THEN LET f=0
109 IF I=0 THEN PRINT AT 5.5,8$
TO 22)
```

Acrescentando ainda esta linha (a seguir), você obterá o movimento do taco:

```

125 PRINT AT 14.3;" ";CHR$ 134;
TAB 2;" ";CHR$ 7;TAB 1;" ";
CHR$ 7;CHR$ 129;TAB 1;" ";CHR$
6;CHR$ 0;CHR$ 6;CHR$ 4;CHR$ 4;T
AB 4;CHR$ 130;CHR$ 0;CHR$ 130;CH
R$ 2

```

SERVICIOS Services Services Servis

**Caixa Postal 62521.
CEP 01150
Tel.: (011)664316
São Paulo-SP**

ACABOU!
VOCÊ VAI ME FALAR
AUTO FELIZ GENIARRA
PARA O COMPUTADOR

E AINDA GANHA O DIREITO DE ADQUIRIR, PELO MENOR PREÇO DO MERCADO, MICROS, SUPRIMENTOS E ACESSÓRIOS. Assomando-se agora, você ainda recebe os 5 boletins já editados pelo clube, incluindo o especial dezembro/85. Solicite, ainda hoje, informações detalhadas acerca de como participar do COMPUCLUB. Não se esqueça, porém, de indicar o tipo de micro que você possui.

COMPUCLUB — Caixa Postal 46 (36570) Viçosa-MG

Rua Antonio das Chagas, 180
CEP 04714 — São Paulo-SP
Tels.: S48-7137 e S23-8S63

JEL PRINT
FORMULÁRIOS E REPRESENTAÇÕES ITOA.

Com esta opção de espaço e custo, ninguém vai deixar de anunciar. Consulte-nos pelo telefone 262-6306 (Departamento de Publicidade), com a Srta. Nilce ou (011) 853-3800, com Srta. Vera.

- Introdução à Microcomputação
- DOS - PC "Sistema Operacional"
- UNIX "Sistema Operacional"
- LINGUAGEM C "Ling. Programação"
- dBASE II "Programação Básica"
- dBASE III "Programação Avançada"
- dBASE III "Programação Básica"
- LOTUS 1-2-3 "Plan. Eletrônica"
- Framework "Sistema Integrado"
- Symphony "Sistema Integrado"
- Wordstar "Processador de Texto"

* Em desenvolvimento

MATERIAIS DIDÁTICOS: Publicações Técnicas desenvolvidas em Português.
RECURSOS DIDÁTICOS: Conceitos e exemplos práticos, através Micros e Têla de 72".
CURSOS FECHADOS E ABERTOS

NOVO ENDEREÇO Rua Groelândia, 1.750 - Jardim Europa
CEP 01434 - São Paulo - SP
Telefones: 881-4599 - 881-4032

- **FITAS**
MAGNÉTICAS
1200 e 2400 pés
- **VHS — BETAMAX**
De T5 a T120
- **DISKETTES**
5 1/4 e 8"

Rua Conde de Leopoldina,
270-A São Cristóvão — R.J.
Tels.: (021) 580-6729
580-8276

- Turmas às 2^{as} e 4^{as}, 3^{as} e 5^{as} e aos sábados.
- Microcomputadores CP-500 da Prológica.
- Plantão de dúvidas.
- Preços promocionais parcelados.
- Apostilas e certificado de conclusão.
- "Clube do Micro" às 6^{as} onde o CP-500 é todo seu;

Rua Machado Bitencourt, 278 (011) 549-9811 S. Paulo - SP
Rua Voluntários da Pátria, 286 - 2º (021) 286-4740 -
R. de Janeiro - RJ

- Micros
- Software
- Assistência Técnica IBM/PC, XT e AT
- Suprimentos
- Acessórios Periféricos

**CONSULTE
NOSSOS PREÇOS!**

LIGUE JÀ!
(011) 276.8988

 **DATAROAD**
Rua Luiz Goés, 1894 — São Paulo
CEP 04043 — Telex: (011) 37755 DTRD

Crie seus próprios jogos e movimente figuras em alta resolução, nos micros da linha Apple, utilizando as rotinas da ROM.

Alta resolução em Assembler (I)

Evandro Mascarenhas de Oliveira

De maneira análoga ao publicado em MS nº 27, serão discutidos os aspectos ligados ao uso das sub-rotinas da ROM e endereços da página zero para a geração e movimentação de figuras em alta resolução gráfica, utilizando a linguagem de máquina do microprocessador 6502.

A alta resolução gráfica ocupa duas páginas de 8 Kb da memória de vídeo, indo a página 1 do endereço \$2000 ao endereço \$3FFF e a página 2 do endereço \$4000 ao endereço \$5FFF.

A página do modo texto é constituída por 24 linhas e 40 colunas, tendo como unidade gráfica a matriz de 7 colunas por 8 linhas. Em modo gráfico de alta resolução, esta matriz é distribuída nas 24 linhas e 40 colunas resultando em $24 \times 8 = 192$ linhas em modo gráfico completo, ou $20 \times 8 = 160$ linhas em modo gráfico/texto e $40 \times 7 = 280$ colunas, num total de $280 \times 192 = 53760$ pontos.

Para diminuir a quantidade da memória de vídeo, relativa aos 53760 pontos, adotou-se um byte de 7 bits (o oitavo bit serve para controle da cor) para unidade gráfica por linha, ficando os 280 pontos em cada linha referenciados por $280/7 = 40$ bytes ou endereços, havendo no total das linhas $192 \times 40 = 7680$ endereços (os 512 endereços restantes não utilizados na memória de vídeo dão o total de 8192 bytes ou 8 Kb por página).

Quando um bit tem o valor 1, acende um ponto na tela; e quando tem o valor 0, mantém-se apagado. Se um determinado byte na memória de vídeo, por exemplo, tem o valor \$73 (binário 01110011), ter-se-á cinco pontos com nível alto — portanto, acesos — e dois com nível baixo — apagados — que se apresentam de maneira reversa ao binário (figura 1).

A determinação da posição de um ponto luminoso e a página da memória de vídeo onde está sendo exibido, estão relacionadas às seguintes sub-rotinas e endereços da página zero:

1 — HPOSN: Sub-rotina com início em \$F411, dá a posição inicial da linha para o ponto de coordenadas C (coluna) e L (linha), sendo que:

C — Varia de 0 a 279 (\$00 a \$117), cujos valores necessitam ser referidos por dois endereços, LSB e MSB, respectivamente,

byte menos significativo e byte mais significativo.

L — Varia de 0 a 191 (\$00 a \$BF), bastando um só endereço para ser referenciada.

Carregando o acumulador com o valor da linha, o registrador X com o LSB e o registrador Y com o MSB da coluna, ter-se-á o endereço inicial da linha nos endereços \$27(MSB) e \$26(LSB), designados, respectivamente, por HBASH e HBASL.

2 — HNDX: Representado pelo endereço \$E5, é o número de bytes a ser adicionado ao endereço inicial da linha. Indica ainda a parte inteira da coordenada (coluna) dividida por sete.

3 — HPAG: Dado pelo endereço \$E6, mostra a página da memória de vídeo onde está sendo exibido o ponto. Tem o valor \$20, se for página 1; e \$40, caso seja página 2.

4 — HMASK: Colocado no endereço \$30, indica a posição do bit ligado (valor 1) dentro do byte das colunas de 0 a 279. O byte zero, representativo das colunas de 0 a 6 (7 colunas),

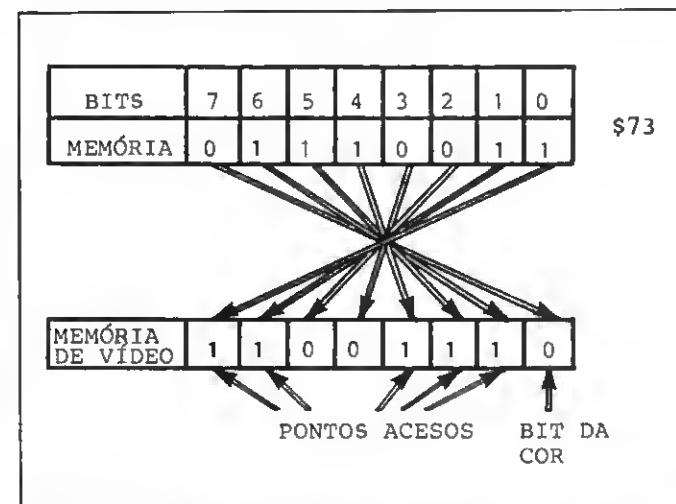


Figura 1 — Hexadecimal 73 e seu respectivo binário em algum endereço da memória, além de sua referência na memória de vídeo.

NA CLAPPY, CRAFT II PLUS E CRAFT-XT COM 3 SOFTWARE EXCLUSIVOS.

CRAFT II PLUS

CPU com teclado numérico reduzido. Monitor profissional Compo (24 MHz). Interface e drive 5' 1/4". 32 funções pré-pro-

gramadas e 10 funções programáveis. Sistema Ivanita (geração de caracteres em português). Conectores banhados a ouro.



CRAFT-XT

PC 16 bits IBM compatível. Memória RAM de 256 kbytes, expandível até 640 kbytes. Opera em tempo real. 2 drives de 5' 1/4" de 340 kbytes cada ou drive de 5' 1/4" de 340

kbytes e até 2 discos rígidos Winchester de 10 Mbytes cada. Teclado de 85 teclas com microprocessador. Monitor monocromático de 12", com tela verde anti-reflexiva.

Na Clappy, o Craft II Plus e o Craft-XT ainda têm 3 software exclusivos: o atendimento especializado, a assistência técnica perfeita e as melhores condições de preço e pagamento da cidade.

Passe na Clappy e faça um teste sem compromisso. Você vai ver como o Craft II Plus e o Craft-XT são altamente compatíveis com você.

MICROCRAFT
MICROCOMPUTADORES LTDA.

Clappy

O lado gente da máquina

Centro: Av. Rio Branco, 12 - loja e sobreloja
Tel.: (021) 253-3395

Copacabana: Rua Pompeu Loureiro, 99
Tels.: (021) 236-7175/257-4398

Visite nossas lojas ou solicite a visita de um representante.



Envie suas dicas para a Redação de MICRO SISTEMAS na Av. Presidente Wilson, 165 - grupo 1210. Centro, Rio de Janeiro, RJ. CEP 20030

Linha SHARP PC-1500

Utilizando rótulos

Facilite a estruturação, a compreensibilidade e a manutenção de seus programas no PC-1500, fazendo uso de rótulos como no exemplo abaixo:

```
Ao invés de:
50 GOSUB 1000
1000 C = P * N: RETURN
```

```
Faça:
50 GOSUB "ROTINA DE
CUSTO"
1000 "ROTINA DE CUSTO"
C = P * N: RETURN
Também aceitam rótulos os
seguintes comandos:
GOTO, IF... THEN, LIST,
LLIST, RESTORE, RUN.
```

Aldo Rodrigues da Costa — MG

Linha APPLE

Mensagem em parábola

Esta dica serve para mostrar no vídeo a mensagem contida na variável A\$. Caso a mensagem exceda os limites da tela, basta modificar a fórmula da linha 30.

```
5 HOME: INPUT "HOME ?": AS$=HOME
10 FOR A=0 TO 6.2 STEP .2
20 B=SIN(A)
30 PRINT TAB(B*10+15): AS
40 NEXT A
50 GOTO 10
```

Jorge Pablo Zapata — BA

Linha TK 2000

Amplie caracteres

Com esta dica, você poderá ampliar os caracteres do seu TK 2000. Digite as linhas abaixo e rode com RUN, depois apague-os:

```
1 DATA 165,9,162,7,173,0,64,133,
0,6,0,6,0,144,5,165,2,32,0,248,1
36,282,288,243,238,2,173,0,3,24,
185,4,141,6,3,281,64,288,217,96
2 FOR F=768 TO 897:READ P:POKE F
,P:NEXT F
```

Para usar o programa, digite:

```
10 HOME: INPUT "QUAL A COR ?
(1=VERMELHO, 2=AMARELO, 3=VERDE, 4=AZUL, 5=ROSA, 6=PRETO)
": AS$
20 GR:POKE 48,A:HTAB 1:VTAB 1:PRI
NT AS$
30 POKE 774,32:POKE 2,10
40 CALL 768
50 GET AS$:TEXT:RUN
```

William Joseph Watterson — SP

Linha ZX Spectrum

Memória disponível

Para saber quanto resta de memória em seu micro, digite as linhas abaixo e rode o programa. Depois, sempre que quiser saber quanto sobra de memória, use PRINT USR 23300.

A rotina fica guardada na área do buffer da impressora, e se você quiser colocá-la em outro local da memória, basta mudar os endereços da linha 10 antes de rodar o programa.

```
10 FOR F=23300 TO 23310
20 READ A:POKE F,A:NEXT F
30 DATA 205,26,31,42,178,92,23,7
,66,68,77,201
```

Carlos Alberto Guerreiro — MG

Linha ZX81

Caleidoscópio

Aí vai uma Dica que explora a capacidade gráfica do seu compatível com ZX81. Após rodar o programa, mude o seu efeito alterando a linha 30, para:

```
30 LET AS=CHR$(INT (RND*10+128))
30 LET AS=CHR$(INT (RND*10+128)+
CHR$(INT (RND*10+1))
```

Podendo então alterar o comprimento da variável A\$.

```
1 REM -- CALEIDOSCOPIO --
10 CLS
20 LET A=1
30 LET AS=CHR$(INT (RND*10+1))
40 LET B=20
50 LET C=8*RND
60 LET D=8*RND
70 PRINT AT C,D:AS
80 PRINT AT D,C:AS
90 PRINT AT C,D:AS
100 PRINT AT B,D,C:AS
110 PRINT AT D,B,C:AS
120 PRINT AT B,C,D:AS
130 PRINT AT C,B,D:AS
140 PRINT AT B,D,B:AS
150 LET A=A+1
160 IF A=20 THEN RUN
170 GOTO 50
```

José Rogério Prates — SP

Linha APPLE

Impressão profissional

Esta dica serve para escrever em seus joguinhos ou aplicativos pequenas mensagens com mais elegância. A mensagem deve ser introduzida na variável A\$, que se encontra na linha 5. Para mudar a linha, onde vai ser impressa a mensagem, basta mudar o VTAB da linha 30.

```
5 HOME: INPUT "FRASE ?": AS$=HOME
10 FOR A=1 TO LEN(AS)
20 VTAB 1:HTAB A
40 BS=MID$(AS,A,1)
45 SOUND ASC(BS)*20,4
50 PRINT BS:CHR$(242):CHR$(199)
60 NEXT A
```

Jorge Pablo Zapata Rivera — BA

Linha ZX Spectrum

UDG 2 do TK90X

A função UDG 2 do TK90X é uma *mão na roda* para redefinir os caracteres do micro, mas nem tudo foi dito no manual. Experimente apertar uma tecla de A a U, acompanhada de CAPS SHIFT e você verá que pode passar a redefinição da tecla pressionada para a letra que está em edição.

Ricardo Costa — RJ

Linha ZX81

Efeito na tela

Esta rotina provoca nos computadores da linha ZX81 um efeito muito interessante no vídeo, podendo ser usada também em qualquer programa:

```
10 REM .....15 caracteres
quaisquer)
20 POKE 16514,62
30 POKE 16515,48
40 POKE 16516,237
50 POKE 16517,71
60 POKE 16518,201
70 RAND USR 16514
80 FOR A=0 TO 100
90 POKE 16515,A
100 RAND USR 16514
110 NEXT A
```

Você notará que após rodar a rotina, todos os caracteres estarão indefinidos; não se alarme, para voltar ao normal é só dar estes comandos diretos:

```
POKE 16515,30 (new line) e
RAND USR 16514
```

Quem Woo Moo — SP

Linha APPLE

Conversão DEC-HEX

Muitas vezes, se faz necessária uma rápida conversão Decimal-Hexadecimal. Normalmente, um programa para esse fim, além de ser lento, necessita de uma série de manipulações de "Strings" e é um pouco dispendioso.

Mas, neste programa, a conversão é feita usando uma rotina intrínseca do Monitor, o que nos garante uma alta velocidade e uma simplicidade maior.

Esta rotina "PRINTA" o conteúdo

dos registros XY em HEXA, e seu endereço é \$FD99.

Observação: O valor máximo decimal é 65535.

```
10 HOME
20 REM DEC-HEX
30 PRINT "ENTRE COM O NUMERO DEC
": INPUT Z:PRINT "PRINT
40 K=INT(Z/256):F=Z-K*256
50 X=F:Y=K
60 POKE 768,162:POKE 769,X:POKE
770,160:POKE 771,Y
70 POKE 772,32:POKE 773,153:POKE
774,253
75 POKE 775,96
80 CALL 768
90 PRINT:PRINT
100 GOTO 30
```

Alexandre Magno Rodrigues Alves — SP

Linha ZX81

Funções especiais

Use esta dica para descobrir o nome de um programa em seu TK85.

```
10 LET Z=0
20 LET Z$=""
30 DIM AS(100)
40 RAND USR 8385
50 FOR I=1 TO LEN AS
60 IF CODE AS(I)<128 THEN NEXT I
70 PRINT "PROGRAMA: ";AS(1 TO I)
```

Enrique Raul Broens — PR

Linha TRS-80

Destrave o LIST no TRS-80

Com o seguinte POKE, qualquer programa com LIST todo bagunçado poderá ser listado SOMENTE NA TELA. POKE 17129,239. Se não funcionar, desligue e ligue o computador, recarregue o programa e dê POKE 17130,66. Se ainda não funcionar, desligue e ligue o computador e vá direto para o BASIC residente, dê um PRINT PEEK (17129), PEEK (17130), anotando os valores impressos e, depois, entre no DOS. Carregue o mesmo programa e dê POKE 17129, n1: POKE 17130, n2, sendo n1 e n2 os valores que você tinha adotado.

Bruno Lopes Fernando Cabral — PB

Linha APPLE e TK 2000

Gravação de dados

Um dos grandes problemas do Applesoft/BASIC é a gravação de dados alfanuméricos em fita, o que dificulta muito a criação de programas do tipo Banco de Dados, por exemplo.

A solução é simular uma função que execute com precisão e segurança. É o que as rotinas abaixo propõem.

ROTINA 1:

```
10 HOME
20 CLEAR
30 DIM A(255):AS(255)
40 INPUT "TEXTO: ";B$
50 A(0)=LEN(B$)
60 FOR I=1 TO A(0)
70 AS(I)=MID$(B$,I,1)
80 NEXT I
90 FOR I=1 TO A(0)
100 A(I)=ASC(AS(I))
110 NEXT I
120 PRINT "PRONTU?":GET X$
130 STORE A
140 PRINT:PRINT "OK"
```

ROTINA 2:

```
10 HOME
20 CLEAR
30 DIM A(255):AS(255)
40 PRINT "PRONTU?":GET X$
50 RECALL A
60 FOR I=1 TO A(0)
70 AS(I)=CHR$(A(I))
80 NEXT I
90 PRINT "TEXTO: ";
100 FOR I=1 TO A(0):PRINT AS(I):
NEXT I
```

Estes dois pequenos programas rodam sem problema no TK 2000. É ACONSELHÁVEL, porém, que estes sejam depurados para funcionarem perfeitamente, de acordo com as necessidades de cada um.

Carlos Henrique C. Duarte — MG

Linha ZX81

INKEY\$ para várias teclas

Diversas vezes, num programa em BASIC, temos necessidade de interpretar mais de uma tecla, pressionada simultaneamente.

Para resolver tal problema, pode ser utilizada uma variável do sistema contida nos endereços 16421 e 16422, consequentemente, o valor das teclas pressionadas, será:

```
PEEK 16421+256*PEEK 16422
```

Basta guardar o resultado em uma variável e, a partir daí, fazer todo o resto. Veja um exemplo com este pequeno programa que interpreta apenas o pressionar simultâneo de 5 e 0 ou 8 e 0, muito utilizado em alguns jogos.

```
10 CLS
20 LET X=PEEK 16421+256*PEEK 16422
30 IF X=56807 THEN PRINT AT 10,16;"5
E 0"
40 IF X=62959 THEN PRINT AT 10,16;"8
E 0"
50 GOTO 10
```

Os valores das teclas pressionadas podem ser obtidos a partir da seguinte rotina:

```
9000 PRINT AT 21,0:PEEK 16421+256*PEEK
16422
9010 GOTO 9000
```

Linha TK 2000

Chamadas em Assembler

Duas dicas para os novos usuários do TK 2000 que acham que call's no TK 2000 não podem ser usados com os do Apple. É um engano. Por exemplo, digite a seguinte rotina:

```
50000 DATA 160,1,162,0,138,24,23
3,1,208,252,141,48,197,232,224,2
55,208,24,1,236,208,237,96
50010 FOR A=768 TO 769:READ B:PO
KE A,B:NEXT
50020 FOR A=1 TO 255 STEP 3:CALL
7A:NEXT
```

E veja que, tanto no Apple quanto no TK 2000, ela produz um interessante efeito sonoro. Portanto, POKEs a partir do endereço 768 rodam perfeitamente no TK 2000.

Para simular o CALL -958 do Apple, basta substituí-lo por PRINT SPC (39): HTAB 1. Os CALL -936 do Integer BASIC de alguns Apple II, também rodam perfeitamente no TK 2000, e sem modificações.

Bruno Lopes Fernando Cabral — PB

Para estudar FORTH, é preciso conhecer, pelo menos um pouco, sobre programação estruturada. Nesta quarta lição, você vai aprender algumas estruturas de controle que FORTH oferece.

Curso de FORTH (IV)

Antonio Costa

Quando FORTH foi inventado, um modismo denominado programação estruturada estava entusiasmando os cientistas de computação. Este modismo influenciou de tal forma as pessoas que conceberam FORTH que se tornou impossível estudar a linguagem sem saber o que é programação estruturada.

Existem trechos de programas que, dependendo da situação, podem ser ou não executados. Outros trechos devem ser executados repetidamente enquanto certas condições forem satisfeitas. Isso significa que toda linguagem precisa ter comandos para controlar a execução dos outros comandos. Tradicionalmente, tal controle era utilizado pelo IF...THEN GOTO. Foi então que o holandês Dijkstra escreveu um artigo denominado "GO TO Considerado Prejudicial". Neste artigo, ele propõe que se evite o uso de GO TO em certas circunstâncias. Para substituir o GO TO, foram propostas estruturas de controle fixas que no FORTH são representadas por IF ELSE THEN, BEGIN UNTIL e BEGIN WHILE REPEAT. Vamos dar uma olhada em cada uma delas.

IF THEN ELSE

O funcionamento do IF pode ser descrito assim: uma comparação é realizada por palavras do tipo > (maior), < (menor) ou = (igual). Esta comparação é feita entre o segundo elemento da pilha e o topo, tendo como resultado os valores booleanos falso ou verdadeiro. Falso é representado por 0 e verdadeiro, por 1. Caso o resultado seja verdadeiro, todas as palavras entre IF e ELSE são executadas. Caso seja falso, as palavras entre ELSE e THEN são executadas.

Existe a possibilidade de IF ser usado sem ELSE. Quando isto ocorre, as palavras que estão entre o IF e o THEN devem ser executadas, se o resultado da comparação for verdadeiro.

A figura 1 resume o comportamento do IF. Vamos ver alguns exemplos que tornarão as coisas mais claras.

Digamos que eu queria definir uma palavra que imprima uma advertência, se o número que está no topo da pilha for maior que 100. Posso fazer isto com a definição abaixo:

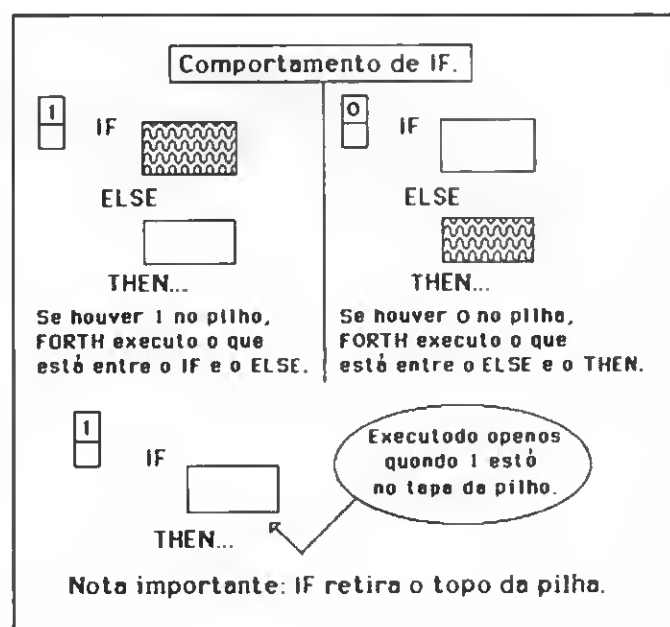


Figura 1

```
: ADV DUP 100 >
  IF "NÚMERO MUITO GRANDE" CR
  THEN ;
```

O DUP serve para fazer uma cópia do número que está no topo da pilha, conforme mostrado no seguinte esquema:



A cópia é necessária porque as palavras comparadoras retiram da pilha os dois elementos que comparam e não queremos perder o número original. O ponto seguido de aspas imprime uma mensagem. No exemplo, esta mensagem é NUMERO MUITO GRANDE. Observe que a mensagem termina com as-

pas. Observe também que não existe espaço entre o ponto e a primeira aspa e que existe um espaço entre a primeira aspa e a mensagem. A palavra CR faz com que o computador mude de linha após a impressão da mensagem. Vejamos o funcionamento da palavra que acabamos de definir:

```
54 ADV OK
200 ADV
NÚMERO MUITO GRANDE OK
```

O número 54 não provocou nenhuma reação de ADV. O número 100 fez com que a mensagem fosse impressa.

Tentemos outro exemplo. Vamos definir uma palavra que substitui o topo da pilha por 1, se ele for maior que 0; por -1, se ele for negativo e por 0, se ele for nulo. Esta palavra é chamada de sinal pelos matemáticos.

```
: SINAL DUP 0 >
  IF DROP 1
  ELSE 0 =
    IF 0
    ELSE -1
  THEN THEN ;
```

Deixo para o leitor a decifração do funcionamento de SINAL.

É boa prática de programação colocar os componentes de uma estrutura de controle um embaixo do outro. Isto pode ser visto na definição de SINAL: os ELSEs, estão embaixo dos IFs, e os THENs, dos ELSEs.

BEGIN UNTIL

O comportamento de BEGIN UNTIL pode ser resumido assim: entre o BEGIN e o UNTIL existem várias palavras que são executadas repetidamente até que o UNTIL encontre a indicação de verdadeiro na pilha. Este comportamento é esquematizado na figura 2. A indicação de verdadeiro, conforme já vimos, é o número 1.

Vamos usar BEGIN UNTIL para definir uma palavra muito útil. Trata-se da SN. Esta palavra lê o teclado até que uma pessoa tecla S (simbolizando sim) ou N (representando não). Caso a letra teclada seja S, 1 é deixado na pilha. Caso seja N, 0 é deixado na pilha. Veja o exemplo:

```
: SN BEGIN "Tecla S(sim) ou N(ão)" CR
  KEY DUP 78 =
  IF DROP 0 1
  ELSE 83 =
    IF 1 1
    ELSE 0
  THEN THEN
  UNTIL ;
```

Os números 78 e 83 são os códigos de S e de N. Se a letra teclada for N (código 73), será executado "IF DROP 0 1". O 1 servirá para indicar ao UNTIL que ele pode parar de pedir à pessoa que

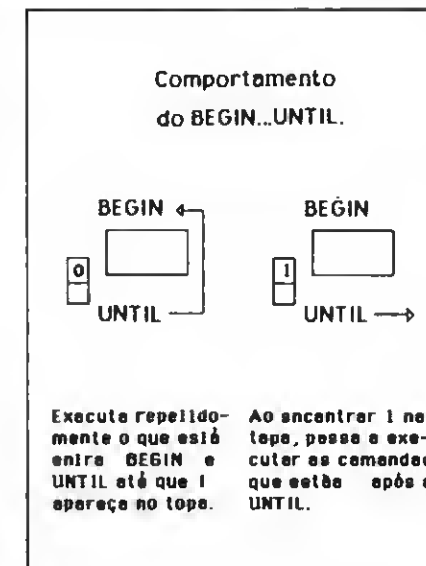


Figura 2

tecle S ou N. O 0 indica que N foi teclado. Se a tecla pressionada não for N, ela é comparada com S (código 83). Caso ela seja a S, FORTH executará "IF 1 1". O primeiro 1 serve para mostrar que S foi apertado; e o segundo, para fazer o UNTIL interromper as repetições. Se a pessoa não apertou nem S nem N, 0 é deixado na pilha e, ao ser encontrado por UNTIL, fará com que o programa peça mais uma vez que o usuário tecla S(im) ou N(ão). Nosso programinha é teimoso: enquanto alguém não aperta S ou N, ele não pára de insistir.

BEGIN WHILE REPEAT

O BEGIN WHILE REPEAT não é muito diferente do BEGIN UNTIL. Entre o BEGIN e o REPEAT há várias palavras que devem ser executadas repetidamente enquanto o WHILE encontrar um sinal de verdadeiro deixado na pilha por alguma comparação. Se WHILE encontrar um sinal de falso, ele interromperá as repetições. O exemplo que vamos fornecer é uma palavra que pede repetidamente números e os soma ao topo da pilha.

```
: SOMATÓRIA 0
  BEGIN "DE-ME UM NÚMERO" CR #IN
  DUP 0 >
  WHILE REPEAT DROP ;
```

Inicialmente, um 0 é colocado no topo da pilha. Números são pedidos, enquanto a pessoa que os está teclando não entrar com um que seja negativo. A palavra que lê inteiros do teclado e os coloca na pilha é # IN.

CARACTERES GRÁFICOS

A maioria dos computadores possui certos códigos que, impressos com

CENTRALDATA
Com e Representações Ltda

SUPRIMENTO É COISA SÉRIA

- DISKETES: 5 1/4 e 8" e fitas magnéticas - marca OATALIFE VERBATIM
- ETIQUETAS PIMACO - PIMATAB
- FORMULÁRIOS CONTÍNUOS E PASTAS
- FITAS P/IMPRESSORAS EM GERAL
- ARQUIVOS PARA DISKETTES

PRODUTOS COM GARANTIA E ENTREGA IMEDIATA

AV. PRES. VARGAS Nº 482 GR. 201/203
Tel.: KS (021) 253-1120
Telex: (021) 34318

EMBLER (I) — Artigo de Evandro a linha Apple.

Série de autoria do professor

53 LIVROS

54 DICAS

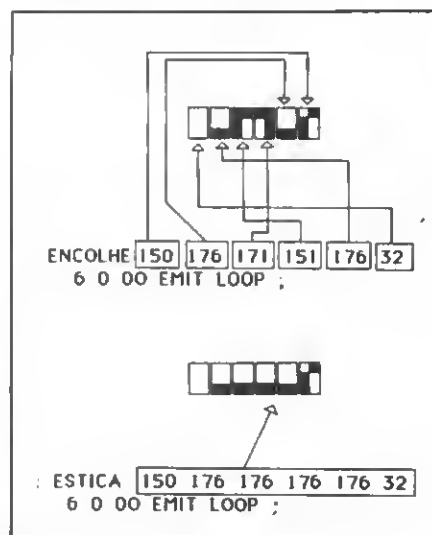


Figura 3

EMIT, produzem símbolos gráficos. É possível imprimir seqüências destes símbolos gráficos de modo a formar desenhos muito úteis na confecção de jogos. A figura 3 mostra uma das muitas maneiras de fazer isto. Nesta figura, há um esquema indicando como uma pequena minhoca egípcia pode ser desenhada tanto com o corpo encolhido

quanto com o corpo esticado. Os códigos gráficos usados para criar a minhoca são de máquinas compatíveis com o TRS-80. Caso sua máquina seja da linha Sinclair, consulte o manual para ver quais símbolos deve usar a fim de obter efeito semelhante.

Na figura 4 mostramos como movimentar a minhoca pela tela. Nesta listagem, a palavra PAUSA emite uma nota musical de certa duração e tem duas finalidades: produzir som e fazer FORTH perder tempo de modo que o movimento da minhoca não seja muito rápido. Observe que usamos a palavra NOTE, cuja definição foi dada na primeira lição. Reutilizar palavras definidas anteriormente é muito popular entre programadores FORTH, pois diminui em muito o tempo de desenvolvimento de aplicativos. Aconselho o leitor a manter uma biblioteca de palavras reutilizáveis.

Vamos agora discutir a definição de MINHOCA apresentada na figura 4. A palavra PAGE apaga a tela. A seqüência "15 EMIT" elimina o cursor de máquinas como o TRS-80 e o CP 500. Caso sua máquina não seja uma destas duas, não é preciso incluir "15 EMIT" na definição. A seqüência "6 I PTC ESTICA" coloca o cursor na coluna I da linha 6 e, em seguida, desenha a minhoca esticada.

```
: ENCOLHE 150 176 171 151 176 32
  6 0 00 EMIT LDDP ;
: ESTICA 150 176 176 176 176 32
  6 0 00 EMIT LDDP ;
: PAUSA 98 40 NOTE ;
: MINHOCA PAGE 15 EMIT
  55 0 00 6 I PTC ESTICA PAUSA
  6 I PTC ENCOLHE PAUSA
  LDDP ;
```

Figura 4

Como I está variando de 0 a 53, haverá a ilusão de movimento.

Gostaria de chamar a atenção do leitor para o fato de que o último caráter do corpo da minhoca é um espaço em branco (código 32, no TRS-80). Se este espaço em branco não for incluído, a minhoca deixará um rastro ao se movimentar.

Antonio Eduardo Costa Pereira é formado em Engenharia Eletrônica pela Escola Politécnica da USP. Fez mestrado em Ciência Especial no Instituto de Pesquisas Espaciais, em S. J. dos Campos, e doutorado em Engenharia Elétrica na Cornell University, de Nova Iorque.

SERVIÇOS

Micro Sistemas

problema

Cr\$ 12.000
pág. Valor

Projeto MICRO BUG

Sim, desejo receber

☐ a fita MICRO BUG, com cartão de referência, pela qual pagarei Cr\$ 40.000 + Cr\$ 7.800 pelas despesas do correio.

Nos atrasados

Sim, desejo receber

☐ os exemplares _____ de MICRO SISTEMAS (do 31 ao 39) pelos quais pagarei o preço unitário de Cr\$ 3.000.

n.º _____ no valor de Cr\$ _____ à ATI Editora.

Cargo _____ Empresa _____

Tel.: _____

Estado _____

ASSINATURA

0030

Rua Oliveira Dias, 153 - CEP 01433
São Paulo/SP - Tel.: (011) 853-7758

SOFT SPEED

GARANTIA TOTAL • MANUAIS INCLUSOS

JOGOS:
ESCOLHA 3
E PAGUE 2 !!!
E AINDA TEM
BRINDE

CP-400 JOGOS EM FITA

F101 - DECATHLON - O melhor tempo de todos
F102 - DOUBLE BECK - Prêmio de imagens
F103 - DRAGONIAN - Salvo o astronauta no espaço
F104 - ICE HOCKEY - Hockey no gelo
F105 - MR. DQ - Fantástico e emocionante
F106 - INTERCEPT 4 - Destrua os inimigos galácticos
F107 - MISTAR - Jogo de aventura
F108 - SIMULADOR DE VOO - Excelente qualidade gráfica
F109 - POPEYE - Derrote o Brutus
F110 - GALAGON - Fantástico jogo
F111 - ZAXXON - O melhor jogo para micros
F112 - GERM - Defesa de dois gêneros
F113 - DEVIL ASSAULT - Emocionante!
F114 - AIR TRAFFIC CONTROL - Controle o aeroporto
F115 - CRASH - Defesa de dois níveis
F116 - SPACE WRECK - Defesa sua base espacial
F117 - SHAFT - Destrua os portos a-muro-cuados!
F118 - DESERT RIDER - Corrida de carros no deserto
F119 - CHOPPER STRIKE - Aventura com helicóptero
F120 - TIME PATROL - Salve os pára-queixas

PREÇO DE CADA JOGO: Cr\$ 35.000

PARA CADA 2 JOGOS COMPRADOS, ESCOLHA UM TERCEIRO INTERAMENTE GRÁTIS. E AINDA GANHE MAIS UM JOGO: BLACKJACK.

CP-400 APLICATIVOS E UTILITÁRIOS EM DISCO

D101 - VIDEALC - Planilha eletrônica 250.000
D102 - VIDEALC - Planilha eletrônica 250.000
D103 - VIDEALC - Planilha eletrônica 250.000
D104 - VIDEALC - Planilha eletrônica 250.000
D105 - VIDEALC - Planilha eletrônica 250.000
D106 - VIDEALC - Planilha eletrônica 250.000
D107 - VIDEALC - Planilha eletrônica 250.000
D108 - VIDEALC - Planilha eletrônica 250.000
D109 - VIDEALC - Planilha eletrônica 250.000
D110 - VIDEALC - Planilha eletrônica 250.000
D111 - VIDEALC - Planilha eletrônica 250.000
D112 - VIDEALC - Planilha eletrônica 250.000
D113 - VIDEALC - Planilha eletrônica 250.000
D114 - VIDEALC - Planilha eletrônica 250.000
D115 - VIDEALC - Planilha eletrônica 250.000
D116 - VIDEALC - Planilha eletrônica 250.000
D117 - VIDEALC - Planilha eletrônica 250.000
D118 - VIDEALC - Planilha eletrônica 250.000
D119 - VIDEALC - Planilha eletrônica 250.000
D120 - VIDEALC - Planilha eletrônica 250.000
D121 - VIDEALC - Planilha eletrônica 250.000
D122 - VIDEALC - Planilha eletrônica 250.000
D123 - VIDEALC - Planilha eletrônica 250.000
D124 - VIDEALC - Planilha eletrônica 250.000
D125 - VIDEALC - Planilha eletrônica 250.000
D126 - VIDEALC - Planilha eletrônica 250.000
D127 - VIDEALC - Planilha eletrônica 250.000
D128 - VIDEALC - Planilha eletrônica 250.000
D129 - VIDEALC - Planilha eletrônica 250.000
D130 - VIDEALC - Planilha eletrônica 250.000
D131 - VIDEALC - Planilha eletrônica 250.000
D132 - VIDEALC - Planilha eletrônica 250.000
D133 - VIDEALC - Planilha eletrônica 250.000
D134 - VIDEALC - Planilha eletrônica 250.000
D135 - VIDEALC - Planilha eletrônica 250.000
D136 - VIDEALC - Planilha eletrônica 250.000
D137 - VIDEALC - Planilha eletrônica 250.000
D138 - VIDEALC - Planilha eletrônica 250.000
D139 - VIDEALC - Planilha eletrônica 250.000
D140 - VIDEALC - Planilha eletrônica 250.000
D141 - VIDEALC - Planilha eletrônica 250.000
D142 - VIDEALC - Planilha eletrônica 250.000
D143 - VIDEALC - Planilha eletrônica 250.000
D144 - VIDEALC - Planilha eletrônica 250.000

PREÇO DE CADA JOGO: Cr\$ 35.000
PARA CADA 2 JOGOS COMPRADOS, ESCOLHA UM TERCEIRO INTERAMENTE GRÁTIS. E AINDA GANHE MAIS UM JOGO: SENHA.

D145 - TAPETE - Copa desportiva e lida disco 300.000
D146 - DISK UTILITY - Copia, backup, desinstalação, endereços, programas, lista arquivos, etc. 200.000
D147 - DISK DRIVER ANALYSER - Análise hardware 200.000
D148 - COLOR OLIVER - Super simulador de som 300.000
D149 - TITULADOR PARA VT - Gerador caracteres 300.000

BRINDE: NA COMPRA DE QUALQUER PROGRAMA EM DISCO PARA CP400 GRÁTIS UM SUPER SIMULADOR DE VOO

CP-400 UTILITÁRIOS EM FITA

F130 - SUPER SCREEN \$1 - 24 linhas texto graf 40.000

JOGOS EM FITA PARA TX90X

F201 - XADREZ - Em sua melhor versão
F202 - REVERSI - Fantástico jogo Ótimo
F203 - GAMAD - Jogo de labirinto
F204 - ANT ATTACK - Defesa de formigas
F205 - CYBER RATS - Cuidado com os ratos!
F206 - HORACIO E AS ARANHAS - um grande jogo!
F207 - 30 TANKS - Batalha com tanques
F208 - SIMULADOR DE VOO - impressionante
F209 - COORKE - Ajude o coronel
F210 - HORACIO ESQUIANDO - Aventura a cavalo
F211 - PYRAMARAMA - Tente vencer este desafio
F212 - ANDROID 2 - Sensacional!
F213 - PLANETOIDS - Explante missão espacial
F214 - MISSILE - Cuidado com os mísseis
F215 - P&ST - Defesa de navios
F216 - MANG MIER - Conduza Willy nas cavernas
F217 - HUNGRY HORACE - Cuidado com os quadros
F218 - PINGALL - Trapaça e hopper
F219 - SPACE RAIDERS - Defesa seu planeta
F220 - ASTRO BLASTER - Cuidado com os invasores
F221 - JET SET WILLY - Tente fazer Willy correr
F222 - GHOST HUNT - Sensacional versão do Pacman
F223 - CHEQUERED FLAG - Proteja um F1 de verdade
F224 - TENNIS - Excelente versão
F225 - ROBOTICS - Defesa os robôs invasores
F226 - GALACTIC ABDUCTORS - Cuidado! o perigo
F227 - SINUCA - Tente matar todos os bois
F228 - MINED OUT - Cuidado com as minas
F229 - MR. WIMPY - Pegue os ingredientes
F230 - GOLFE - Excelente e real jogo
F231 - CROSS - Tente atravessar a sua perigosa
F232 - JUMPING JACK - Cuidado com as fendas no chão
F233 - GROUND ATTACK - Planete na caverna inimiga
F234 - MONSTER MORE - Cuidado com os monstros da mina
F235 - SEEDS ATTACK - Proteja sua cidade das aves
F236 - GALAXIANS - Destrua os invasores

PREÇO DE CADA JOGO: Cr\$ 35.000
PARA CADA 2 JOGOS COMPRADOS, ESCOLHA UM TERCEIRO INTERAMENTE GRÁTIS. E AINDA GANHE MAIS UM JOGO: SENHA.

APPLE II - JOGOS ESPECIAIS EM DISCO - CADA Cr\$ 120.000

D001 - KARATECA - Luta marcial de verdade
D002 - CONAN - Aventura igual ao filme
D003 - GHOSTBUSTERS - Aventura igual ao filme
D004 - SUMMER GAMES - 8 Modalidades esportivas
D005 - CRI - Jogo pedagógico para adultos
D006 - SKYFOX - Proteja um país de verdade

APPLE II - JOGOS EM DISCO DA PROMOÇÃO

D007 - AZTEC - Defesa a qualquer preço
D008 - PICADILLY - Sensacional jogo
D009 - SHIFT SAM - Fantástico e impressionante jogo
D010 - DONKEY KONG - Mãos corra o gorila
D011 - FREE FALL - Tente vencer este desafio
D012 - PIEMAN - Excelente e divertido
D013 - DRIBITRON - Em e em órbita com seu modo
D014 - CIBERT - Tente se movimentar nos cubos
D015 - JUNGLE HUNT - Aventura na selva
D016 - CAVERNS OF CAUSTO - Aventura
D017 - PITFALL II - Nem cavernas perigosas
D018 - TAINMAN - Labirinto tipo Pacman
D019 - XADREZ - Trapaça e hopper
D020 - GAMAD - Jogo misterioso de labirinto
D021 - BUCK ROGERS - Proteja uma espaçonave
D022 - LOAD RUNNER - Pegue os tesouros

PREÇO DE CADA JOGO: Cr\$ 60.000
PARA CADA 2 JOGOS COMPRADOS, ESCOLHA UM TERCEIRO INTERAMENTE GRÁTIS. E AINDA GANHE MAIS UM JOGO: POKER.

APPLE II - UTILITÁRIOS E APLICATIVOS EM DISCO

D031 - DISK DRIVE DIAGNÓSTICO 200.000
D032 - SAY IT - Simulação de voz 200.000
D033 - MUSICOMP - Composição sua música 200.000
D034 - KODAL - Cuidado com os desenhos 200.000
D035 - PRINT SHOP - Gerador de impressões 170.000
D036 - THE COMPLETE GRAPHICS SYSTEM 200.000
D037 - SCREEN WRITER - Processador de textos 200.000
D038 - SPEED-MALA DIRETA 300.000
D039 - SPEED-CONTROLE DE ESTOQUES 300.000
D040 - SPEED-CONTAS A RECEBER 300.000
D041 - SPEED-FOLHA DE PAGAMENTO 300.000
D042 - LOCKSMITH \$-B - Utilidade de Copia 170.000
D043 - MAGIC WINDOW - Processador de textos 300.000
D044 - VISUALC II - Planilha Eletrônica 170.000
D045 - VISFILE II - Banco de Dados 500.000
D046 - DBASE II - Banco de Dados 450.000
D047 - FRIDAY - Computo 400.000
D048 - WORDSTAR - Processador de Textos 170.000
D049 - LOTERIA ESPORTIVA 170.000
D050 - LOTO 170.000

BRINDE: GRÁTIS, NA COMPRA DE QUALQUER PROGRAMA EM DISCO UM CURSO PASSO-A-PASSO EM BASIC.

COMO COMPRAR:

NÃO RECORTE SUA REVISTA. FAÇA O SEU PEDIDO POR CARTA, RELACIONANDO SEUS DADOS E OS CÓDIGOS DOS PROGRAMAS DESEJADOS. ANEXE À CARTA UM CHEQUE NOMINAL À ATI Editora Ltda., Av. Pres. Wilson, 165 grupo 1210 - Centro, CEP 20.030 - Rio de Janeiro - RJ. As despesas do correio já estão incluídas.

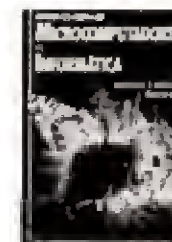
SOFTSPEED - CAIXA POSTAL 40088 - RIO-RJ

ATI EDITORA

NOVOS LANÇAMENTOS



OS 40 MELHORES JOGOS PARA SEU TG-200 e SINCLAIR — Uma sensacional coleção de jogos de inteligência e ação incluídos jogos em linguagem de máquina. Código 200-1. Preço: Cr\$ 29.990.



INTRODUÇÃO BÁSICA AOS MICROCOMPUTADORES E A INFORMÁTICA, de Marcelo Menezes e Renato Audi. Livro que aborda de forma simples e clara a evolução dos computadores, linguagem de programação, atividades profissionais, entre outras assuntos. Código 200-2. Preço: Cr\$ 29.990.

NAS MELHORES LIVRARIAS DO PAÍS

PEDIDOS PARA ATI Editora Ltda., Av. Pres. Wilson, 165 grupo 1210 CEP 20.030 - Rio de Janeiro - RJ, anexando cheque nominal.

A Microdigital lança no Brasil o micro pessoal de maior sucesso no mundo.

A partir de agora a história dos micros pessoais vai ser contada em duas partes: antes e depois do TK 90X.

O TK 90X é, simplesmente, o único micro pessoal lançado no Brasil que merece a classificação de "software machine": um caso raro de micro que pela sua facilidade de uso, grandes

recursos e preço acessível recebeu a atenção dos criadores de programas e periféricos em todo o mundo.

Para você ter uma idéia, existem mais de 2 mil programas, 70 livros, 30 periféricos e inúmeras



revistas de usuários disponíveis para ele internacionalmente.

E aqui o TK 90X já sai com mais de 100 programas, enquanto outros estão em fase final de desenvolvimento para lhe dar mais opções para trabalhar, aprender ou se divertir que com qualquer outro micro.

O TK 90X tem duas versões de memória (de 16 ou 48 K), imagem de alta resolução gráfica com 8 cores, carregamento rápido de programas (controlável pelo próprio monitor), som pela TV, letras maiúsculas e minúsculas e ainda uma exclusividade: acentuação em português.

Faça o seu programa: peça já uma demonstração do novo TK 90X.

MICRODIGITAL

Chegou o micro cheio de programas.

